APPLIED ENGINEERING AND TECHNOLOGY DIRECTORATE

Landsat 7 Processing System (LPS)
Users Guide

Release 3

Volume 1

March 1998

National Aeronautics and
Space Administration —

Goddard Space Flight Center
Greenbelt, Maryland

Landsat 7 Processing System (LPS) Users Guide

Release 3

March 1998

Prepared by:

_____

Neil Ottenstein, Systems Engineering   Date
Landsat 7 Processing System, CNMOS, Computer Sciences Corporation

Reviewed by:

_____

Robert Schweiss, Systems Engineering Manager   Date
Landsat 7 Processing System, Code 586, Goddard Space Flight Center

_____

Edward Criscuolo, Software Task Leader   Date
Landsat 7 Processing System, CNMOS, Computer Sciences Corporation

Quality Assured by:

_____

Sheila Whisonant, Quality Assurance Officer   Date
Landsat 7 Processing System, CNMOS, Computer Sciences Corporation

Approved by:

_____

Nathaniel Daniel, Element Manager   Date
Landsat 7 Processing System, CNMOS, Computer Sciences Corporation

_____

Joy Henegar, Project Manager   Date
Landsat 7 Processing System, Code 586, Goddard Space Flight Center

Goddard Space Flight Center
Greenbelt, Maryland

Preface

This document is under the control of the LPS Project Configuration Management Board (PCMB).

Configuration change requests (CCRs) to this document, as well as supportive material justifying the proposed change, shall be submitted to the LPS PCMB. Changes to this document shall be made by document change notice (DCN) or by complete revision.

Address questions and proposed changes concerning this document to

> Robert Schweiss, Systems Engineering Manager
> Landsat 7 Processing System
> Code 586
> Goddard Space Flight Center
> Greenbelt, Maryland 20771

Abstract

This document provides instructions for operating the Landsat 7 Processing System (LPS). It instructs the novice user in the performance of specific LPS operations and provides a detailed reference for the experienced user.

Keywords: *Landsat 7, Landsat 7 Processing System (LPS), users guide*

Contents

1. Introduction

2. Getting Started

3. Data Capture Operations

4. Raw Wideband Data Management

5. Level 0R Processing Management

6. Files Transfers to EDC DAAC

7. Report Generation

8. Testing the LPS

9. LPS Maintenance

10. LPS GUI Detailed Reference

Appendix A. LPS Error Messages

Appendix B. Directory Structure and File Name Formats

Appendix C. Process Catalog and Man Pages

Appendix D. Customizing Your Environment

Appendix E. Performing LPS Functions Through IRIX

Appendix F. LPS Trouble File Format

Appendix G. Support of DES by the LPS

Abbreviations and Acronyms

Glossary

References

Index

Tables

Figures

1. Introduction

1.1 Purpose and Scope

This document provides instructions for operating the Landsat 7 Processing System (LPS). It instructs the novice user in the performance of specific operations and provides a detailed reference for the experienced user.

This users guide covers the following topics:

- Logging onto the LPS system hosts

- Setting up one's user environment

- Starting up and shutting down the LPS software

- Navigating the LPS graphical user interface (GUI)

- Monitoring LPS status

- Performing basic operations with the LPS

- Detailed LPS GUI menu options reference

- LPS utilities

- LPS troubleshooting

This guide assumes the reader is familiar with point-and-click interfaces that use a mouse, buttons, pulldown menus, and dialogs (or forms) and with the fundamentals of interacting with one of the UNIX shells (csh or sh). This guide also assumes familiarity with LPS standard operating procedures.

This guide does not provide instructions for third-party software such as Oracle's Database Management System (DBMS) or the Generic Telemetry Simulator (GTSIM). The reader should consult vendor-supplied documentation for those tools. This guide also does not provide instructions for LPS software maintenance. The reader should consult the *LPS Programmers Reference Manual* (Reference 1).

1.2 Organization

This document contains 10 chapters and 5 appendixes:

- Chapter 1, *Introduction*, provides an introduction to this document and an overview of the LPS.

- Chapter 2, *Getting Started*, provides instructions for logging on to the LPS, setting up an environment, starting up and shutting down the LPS software, navigating the LPS operations interface, and monitoring LPS status.

- Chapter 3, *Data Capture Operations*, provides instructions for managing and monitoring automatic data capture and for performing data capture manually.

- Chapter 4, *Raw Wideband Data Management*, provides instructions for managing and monitoring the automatic movement of captured raw wideband data to the 30-day store and moving captured raw wideband data files between LPS disks and the removable media of the 30-day store manually.

- Chapter 5, *Level 0R Processing*, provides instructions for managing automatic Level Zero R (L0R) processing, performing L0R processing manually, monitoring L0R processing, and modifying L0R processing parameters and error-reporting thresholds.

- Chapter 6, *File Transfers to EDC DAAC*, provides instructions for managing and monitoring the automatic transfer of LPS output files to the Earth Resources Observation System (EROS) Data Center (EDC) Distributed Active Archive Center (DAAC) and for performing the transfer manually.

- Chapter 7, *Report Generation*, provides instructions for generating LPS reports.

- Chapter 8, *Testing the LPS*, provides instructions for performing LPS built-in tests.

- Chapter 9, *LPS Maintenance*, provides instructions for using assorted LPS utility programs.

- Chapter 10, *LPS GUI Detailed Reference*, provides a detailed explanation of all menu options in the LPS GUI.

- Appendix A, *LPS Error Messages*, describes all error messages output by the LPS.

- Appendix B, *LPS Directory Structure and File Name Formats*, explains the directory hierarchy on LPS strings and the format of LPS file names.

- Appendix C, *Process Catalog and Man Pages*, describes all LPS processes.

- Appendix D, *Customizing Your Environment*, explains what environment variables are important for LPS operations and how to customize their values.

- Appendix E, *Invoking LPS Programs From IRIX*, explains how to execute LPS programs and use IRIX and ORACLE features to perform LPS operations from an IRIX shell. Expert users can use this information to construct scripts that extend the capabilities of the LPS.

- Appendix F, *LPS Trouble File Format*, describes the format of the LPS trouble files.

- Appendix G, *Support of DES by LPS*, describes LPS support of the DAAC Emergnecy System (DES) interface.

1.3 LPS Overview

This section describes the LPS environment, functions of the LPS, its hardware configuration, the components of its software, and the operator's role in LPS operations.

1.3.1 Environment

The LPS is a component of the Landsat 7 System. The Landsat 7 System provides wide-area, multi-spectral imaging of the Earth. Landsat 7 data can be processed into images or used in a variety of scientific, military, and commercial applications. The Landsat 7 System consists of flight components (including the Landsat 7 satellite) and ground components (including the LPS). Landsat 7 flight components are responsible for launching, placing, and maintaining the Landsat 7 satellite. Landsat 7 ground components are responsible for operating the satellite and for receiving, processing, archiving, and distributing Earth observation data. Figure 1–1 shows the elements of the Landsat 7 System that are important to LPS operations.
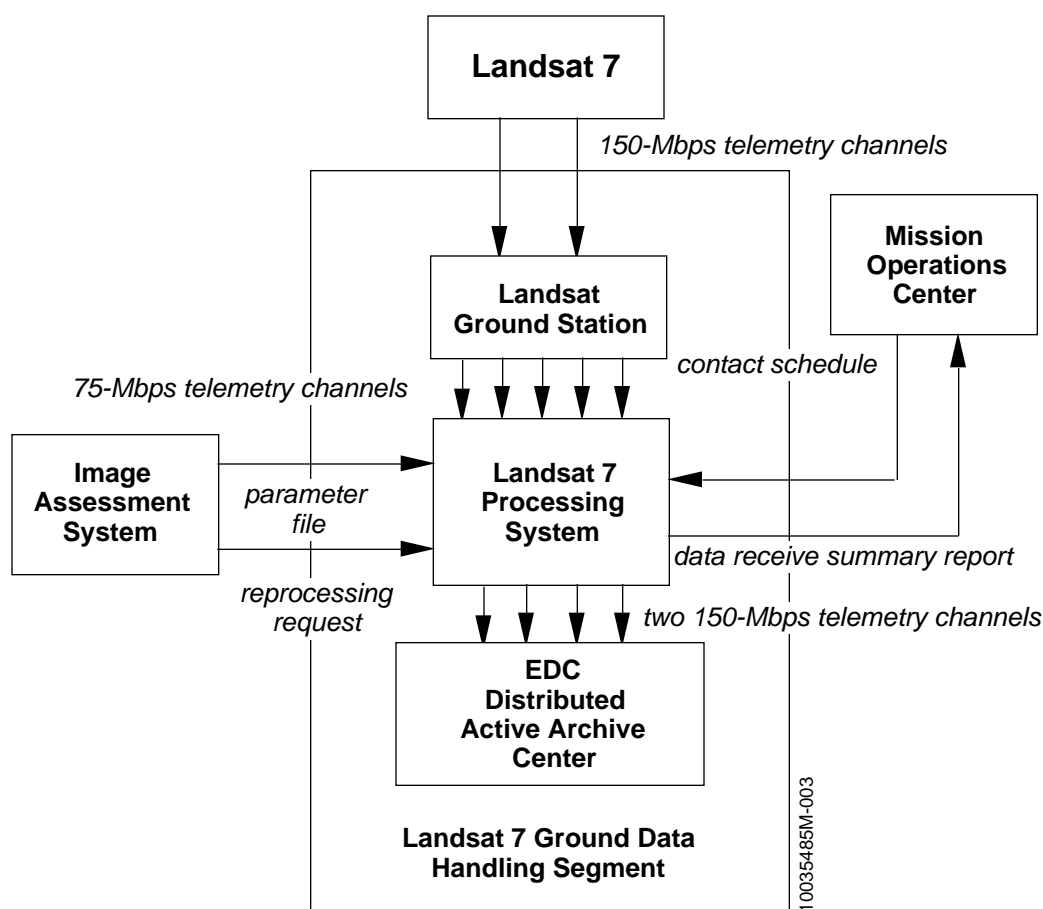
Figure 1–1. The LPS Environment

The Landsat 7 satellite is in a Sun-synchronous orbit along a track defined by the Worldwide Reference System (WRS). The WRS divides the Earth into 57,784 scenes, each representing a swath traversed by the satellite in 23.92 seconds. The Landsat 7 satellite's flight repeats the coverage of all WRS scenes every 16 days.

Onboard the Landsat 7 satellite is the Enhanced Thematic Mapper Plus (ETM+), a fixed-position, nadir-viewing instrument. Its viewing swath is produced by a mirror that sweeps across track as the sensor field of view moves forward due to satellite motion. Calibration data and mirror scan correction data (MSCD) are collected at the end of each scan-mirror cycle. The ETM+ collects data in eight spectral bands. Bands 1, 2, 3, 4, and 8 collect the visible near infrared spectral range. Bands 5 and 7 are short wavelength infrared bands. Band 6 is a thermal long wavelength infrared band.

The ETM+ provides output data via two 75 megabits per second (Mbps) channels. Channel I contains multiplexed data from Bands 1, 2, 3, 4, 5, and 6. Channel Q contains multiplexed data from Bands 6, 7, and 8. Band 6 data is duplicated in both channels. Both channels also contain ETM+ status and payload correction data (PCD) associated with the image data. The Landsat 7 satellite uses one of two onboard solid state recorders to capture imaging data when it cannot be transmitted to the ground in real time.

When a ground station is in sight, Landsat 7 uses two of its three 150-Mbps X-band direct downlinks to simultaneously transmit both real-time and recorded image wideband data to the ground station. Several different segments of image wideband data collected at different times and locations, called subintervals, are normally transmitted in succession. Each segment contains continuous imagery for one or more WRS scenes. Because the satellite can play back any portion of the data recorded in its onboard recorders in any order, the acquisition times and locations of the scenes in the subintervals transmitted during a contact have no particular relation to one another.

The Landsat Ground Station (LGS) acquires ETM+ wideband data directly from Landsat 7 via two 150-Mbps X-band return links, separates each X band into two 75-Mbps I and Q channels, and transmits the acquired wideband data over four 75-Mbps LGS output channels to the LPS. The LGS and LPS receive Landsat 7 contact period schedules from the Landsat 7 Mission Operations Center (MOC). The LPS uses them to schedule the start and stop of its capture of raw wideband data. After each contact, LPS operations personnel generate a Data Receive Summary Report, describing the data captured, and forward it to the MOC, either through voice link or facsimile transmission.

After a contact, the LPS processes the captured raw wideband data to L0R (described in Section 1.3.2) and automatically notifies EDC DAAC that files are available. EDC DAAC transfers the output files and archives them. Landsat 7 data users access EDC DAAC to query the archive for available data and to order data products.

The Image Assessment System (IAS) endeavors to calibrate, maintain, and quantify Landsat 7 data quality. The IAS reviews the quality of a small sample of Landsat 7 data. When problems are discovered, IAS operations personnel may request that data be reprocessed. IAS operations personnel specify the contact period for reprocessing to LPS operations personnel. LPS operators restage the raw wideband data from tape to disk and initiate its reprocessing. The resulting output files are then automatically transferred to EDC DAAC.

1.3.2 Functions

The LPS's principal functions are capturing raw wideband data from the Landsat 7 satellite through the LGS, processing the captured data to L0R, copying the data to Digital Linear Tape (DLT   ), and cooperating in the transfer of the L0R data files to EDC DAAC. The following subsections describe each of these functions in more detail.

1.3.2.1 Raw Wideband Data Capture

The LPS captures raw wideband data transmitted from the Landsat 7 satellite onto a 34-gigabyte (GB) storage array. Each operational LPS string captures one of the four 75-Mbps channels received from the LGS. LPS software captures raw wideband data automatically, based on a contact schedule received from the MOC loaded into the database on operator command. The schedule indicates contact periods for both 150-Mbps downlinks from the Landsat 7 satellite. A capture source field in the LPS database that can be set by an operator indicates the channel to which a string is connected. A daemon process (mac_autocapture) on each LPS string monitors the schedule. It begins data capture shortly before scheduled acquisition of signal for the channel to which the daemon's LPS string is connected. Capture stops automatically at the scheduled time for loss of signal.

Everything the LPS receives through the LGS channel during the scheduled time of contact is written to a standard UNIX file. Because each string creates a separate file, four files are created for each contact. What the LPS receives can include noise as well as actual wideband data. Noise is received whenever there is a dropout (a temporary loss of the satellite's transmission) or when any other error in the communications path occurs causing an incorrect value to be received by the LPS.

When a contact is over, each LPS string automatically copies the newly created raw wideband data file to a DLT™. Because a second contact can occur while one contact's file is being copied to tape, the LPS software will wait for the tape drive to become available and, if necessary, for a new tape to be inserted.

1.3.2.2 L0R Processing

When a contact is over, each LPS string automatically begins processing the raw wideband data to L0R. Because a second (and indeed a third) contact can occur while one contact's file is being processed to L0R, the LPS software will wait until L0R processing on the first contact has completed before beginning work on subsequent contacts.

L0R processing is a multi-step process that locates and extracts ETM+ imagery and ancillary data from the raw wideband data stream captured during a contact to produce a number of output files written to a 34-GB storage array, separate from the storage array used to hold raw data.

The L0R processing steps are as follows:

- *Raw Data Processing* – LPS software extracts transfer frames from the raw wideband data file, applies a number of error detection and correction schemes to remove transmission errors, and annotates the transfer frame with a number of quality indicators before passing it on for further processing.

- *Major Frame Processing* – LPS software reconstructs ETM+ major frames from data extracted from the transfer frames, identifies the subintervals making up the contact, extracts image data for each of the eight spectral bands, and creates output files containing the MSCD and calibration data for each subinterval.

- *Image Data Processing* – LPS software takes the extracted image data and, for each subinterval, creates output files for each spectral band (in the case of Band 8, as many as four output files can be created for a single subinterval), produces browse files containing a reduced resolution image for each scene in the subinterval, automatically assesses the cloud cover for each scene, and provides a Moving Window Display (MWD) of the imagery being processed.

- *PCD Processing* – LPS software reconstructs PCD major frames from data extracted from transfer frames. LPS software also extracts the attitude and ephemeris data contained in each PCD major frame to determine the time at which the satellite passed over WRS scene centers. The scene center times are used to divide the imagery into scenes during image data processing.

- *Metadata File Generation* – LPS software generates a metadata file for each subinterval. The metadata provides information on the identification and quality of the L0R instrument data contained in a subinterval. Metadata also includes return-link data quality and ETM+ instrument data quality statistics, quality and accounting (Q&A) information on the received and processed PCD, and Cloud Cover Assessment (CCA) on both WRS scene and quadrant bases.

1.3.2.3 Output File Transfer

When L0R processing has completed successfully, the LPS software on each string automatically notifies EDC DAAC of its availability by sending a data availability notice (DAN). On receipt of this notification, EDC DAAC acknowledges the DAN by returning a data availability acknowledgment (DAA). Later, EDC DAAC transfers the available files from the LPS to its data archive storage using File Transfer Protocol (FTP). EDC DAAC notifies the LPS that it has attempted to transfer the files by sending a data delivery notice (DDN) describing the disposition of the transfer attempts. The LPS acknowledges the DDN by returning a data delivery acknowledgment (DDA).

If the disposition in a DDN indicates that a file has been successfully transferred, the LPS automatically deletes the file. Otherwise, the LPS outputs messages indicating that the transfer attempt failed. Files can be marked for retention. In this case, the output file is not deleted automatically. Files can be manually deleted at any time as deemed necessary.

1.3.3 Hardware Configuration

The LPS hardware configuration for Release 1 is illustrated in Figure 1–2. The configuration includes the following components:

LPS Strings – These are the hosts for nearly all LPS processing. The host for each string is a Silicon Graphics, Inc. (SGI) Challenge XL computer. Each string has eight
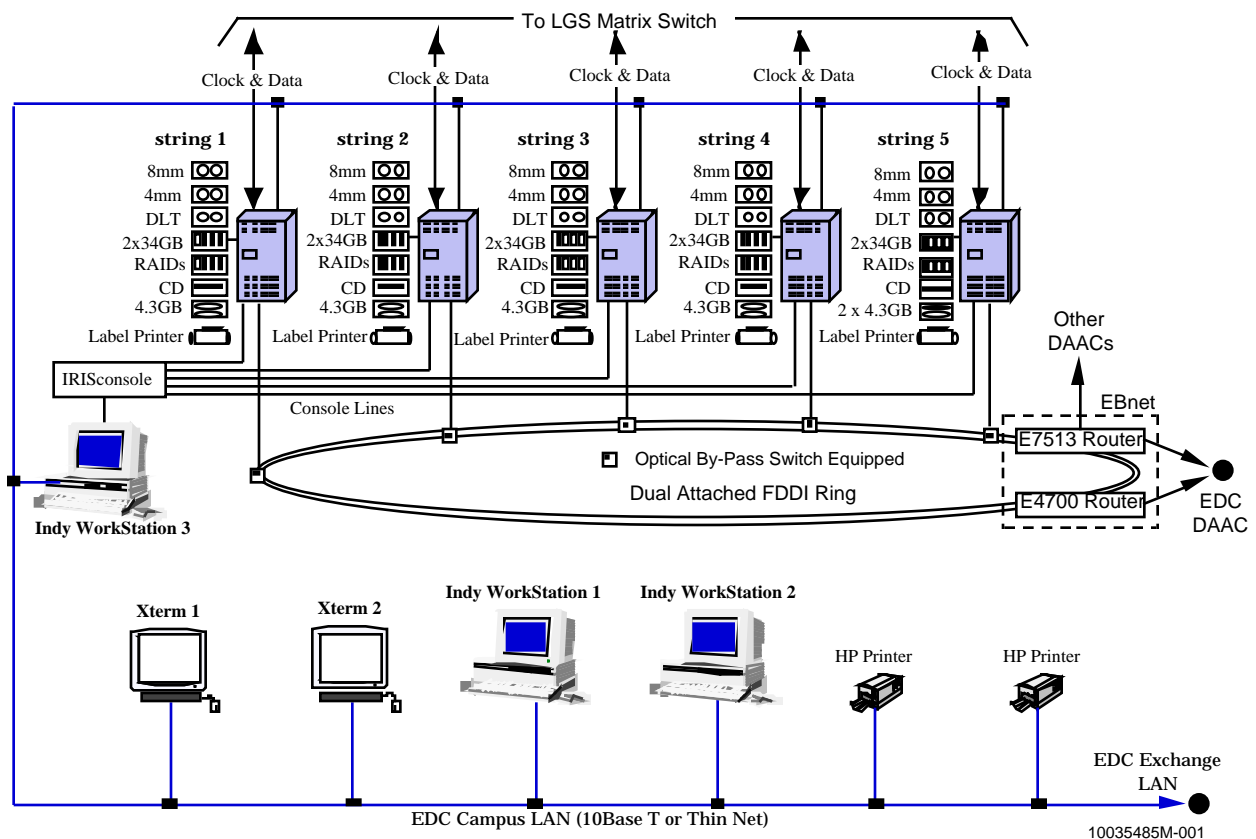
Figure 1–2. LPS Release 1 Hardware Configuration

processors and 512 megabytes of random access memory (RAM). Each string contains LGS output channel interface hardware through which the string can capture raw wideband data and a 34-GB storage array on which it can store captured data. For Release 1, the LGS interface hardware of one LPS string can be connected to any other to simulate the LGS-LPS interface. Another 34-GB storage array holds the output files produced by L0R processing. A 4-GB disk stores all other files, including the LPS executables and the LPS database. Each string hosts a Box Hill MDLIC-7-DLT4 media changer drive used to save raw wideband data to tape and to restage it to disk when reprocessing is requested. Each string also hosts a CD-ROM drive, a 4-millimeter tape drive, and an 8-millimeter tape drive. LPS strings are connected to each other and to three Indy workstations and two X terminals by an Ethernet network. A fiber-distributed digital interface (FDDI) network serves as the interface to EDC DAAC. For Release 1, the FDDI network interconnects the strings so that one can serve as an EDC DAAC simulator. Finally, each LPS string has two serial connections to Iris console device no. 3, which serves as the string's console terminal.

• *Indy Workstations* – SGI Indy workstations are the hosts for the LPS MWD and also host the LPS string console terminals. They may be used to display the LPS software's GUI for one or more strings.

- *X Terminals* – These are the display devices for the LPS software's GUI. Each string has its own GUI, and a single X terminal can display the GUI from any number of strings. Normally, a single X terminal displays the GUI for two strings.

1.3.4 Software Components

LPS functions are performed by a number of software components, most of which are LPS application software written specifically for the LPS. However, a number of components are general-purpose software. The programs fall into three categories:

1. A collection of separate LPS standalone programs that perform specific LPS operations, such as capturing raw wideband data or processing a specified file to L0R

2. Daemon processes that execute in the background to control the automated LPS functions of data capture, copying to tape, and L0R processing

3. A GUI that allows the user to control and monitor the execution of the LPS programs

A basic feature of the LPS software is its division into functional and managing components. The standalone programs (category 1) perform specific LPS operations. The managing components (categories 2 and 3) do not perform operations. Rather, they cause the operations to occur at the appropriate time.

Figure 1–3 illustrates the LPS software's basic structure. Detailed descriptions of LPS processes appear in Appendix C. Once the LPS software has started, the LPS GUI and daemon processes are always active. Each command issued through the LPS GUI causes it to execute one (or more) of the LPS functional programs to perform the operation requested. The daemon processes are always active in the background. Each of them is waiting for an event. When the event occurs, the daemon starts up a functional program to perform the necessary processing. L0R processing is executed by a number of processes. Each process performs one of the steps necessary to transform raw wideband data into LPS output files.

```
                              mac_lps
                                 │
                 mac_ui_xxx.fmx (Forms 4.5 routines)
                                 │
   ┌─────────────┬──────────────┼──────────────┬──────────────┐
   │             │              │              │              │
  rdc      rdc_terminate  mac_mainStartL0R   rcvddn         stopddn
                                 │
           ┌─────────────┬───────┼───────┬──────────────┐
           │             │       │       │              │
          rdp           mfp     pcd     idp          senddan
                                         │
                                      idp_band
```

10035485M-002
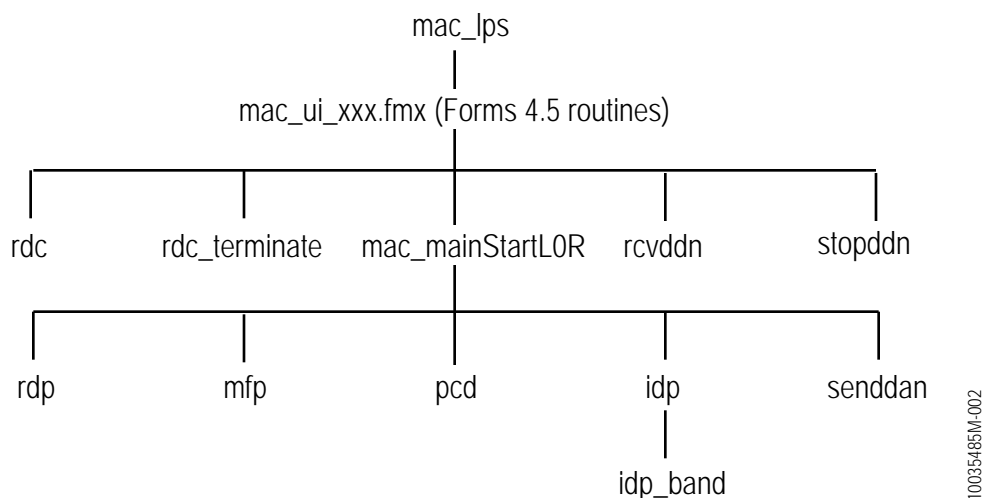
Figure 1–3. LPS Software Structure

LPS functional programs normally are invoked through the LPS GUI. However, all LPS functional programs may also be invoked directly from any IRIX shell (e.g., csh or sh). Invoke them, as for any IRIX program, by typing the program name and any parameters as a command to a shell. Commands can be included in shell scripts.

By writing such shell scripts, users can extend the capabilities of the LPS themselves. For details on invoking LPS functional programs from a shell, read Appendix E.

1.3.5 Operator Role

The LPS operator plays an active role in LPS operations. The following subsections describe how the LPS operator uses LPS software to perform the basic LPS operations of data capture, L0R processing, and output file transfer, as well as operations for testing and maintenance.

1.3.5.1 Data Capture

The LPS operator's role in data capture is to provide contact schedules from the MOC for ingest into the databases on each LPS string; monitor automatic, scheduled data capture; and generate a Data Receive Summary Report after each contact and relay it to the Landsat 7 MOC if needed. In more detail, the LPS operator's role is as follows:

- The MOC will use FTP to put the contact schedules on an Indy workstation. Use IRIX FTP to retrieve contact schedules onto the development string.

- Use the "Ingest Contact Schedule..." command from the LPS GUI to ingest the contact schedule into each LPS string.

- Use the "View/Edit Capture Source..." command from the LPS GUI to set the source of the incoming data each time it changes.

- Use the LPS "Data Receive Summary Report..." command from the LPS GUI to generate a Data Receive Summary Report after each contact for forwarding to the MOC if anomalies are present.

- Monitor the capture of raw wideband data by reviewing the status and error messages generated by the LPS software.

- Ensure that a new tape is available for copying the raw wideband data to the 30-day store and monitor the copying operation by reviewing the status and error messages generated by the LPS software.

1.3.5.2 L0R Processing

The LPS operator's role in L0R processing is to monitor automatic L0R processing, verify L0R processing quality during and after the processing, and perform reprocessing when requested by the IAS. In more detail, the LPS operator's role is as follows:

- Monitor automatic L0R processing by reviewing the status and error messages generated by the LPS software.

- Set up the MWD processing parameters to monitor L0R processing quality while the processing is in progress. This may include specifying an Internet Protocol (IP) address if viewing at a remote station.

- Use the "LPS Q/A Report..." command to generate an LPS Q&A report to review L0R processing quality after processing has completed.

- Service reprocessing requests by using the "Start restage..." command to restage raw wideband data from tape and then use the "Start Level 0R Processing..." command to perform L0R processing.

- Restage data.

1.3.5.3 Output File Transfer

The LPS operator's role in output file transfer is to monitor the automatic transfer of files from the LPS to EDC DAAC. In more detail, the LPS operator's role is as follows:

- Monitor automatic file availability notification and file transfer disposition by reviewing the status and error messages generated by the LPS software.

- Review daily file transfer activity by using the "File Transfer Summary..." command from the LPS GUI to generate a File Transfer Summary Report.

1.3.5.4 Test and Maintenance

In addition to normal LPS operations, the LPS operator uses LPS software to test LPS functions. In more detail, the LPS operator's role is as follows:

- Test LPS data capture functions by using the "Send Test Data..." command on a source LPS string to play back a test data set and use the manual "Start Data Capture..." command on a destination LPS string to capture the data.

- Use the "DAN Transfer State..." command from the LPS GUI to disable automatic file availability notification and then use the "Start Level 0R Processing..." command from the LPS GUI to process a test data set to L0R.

- Monitor L0R quality using the MWD and LPS Q&A report as described in Section 1.3.5.2.

1.4 Notational Conventions

This users guide uses various type styles for different categories of terms. Table 1–1 describes these conventions.

Table 1–1. Text Style Conventions

Text Style	Description
Control>Start L0R ...	Menu items you select from the LPS GUI menu bar or other system menus are shown bold face in Arial font. The description shows the top-level menu name, the symbol ">", and the menu option. Pull-right menus may have additional entries. For example, **Setup>Parameters...>Raw Data Processing...** indicates the Raw Data Processing... menu option, which is in the Parameters... pull-right menu, which is itself under the Setup menu.

```% cat /etc/passwd \```
 	
```? > savefile```
	Any text that you would type as an IRIX shell command appears in Courier font. "%" represents the IRIX command prompt and "?" represents the IRIX continuation prompt. <u>Do not</u> type % or ? when they appear in these positions.
	
	In this document, commands are sometimes split to accommodate the document's format. You can omit the continuation and type the entire command on a single line.

```SQL> SELECT *```
 	
```2 FROM rdc_acct```
 	
```3 WHERE csid = 123;```
	Any text that you would type as input to ORACLE SQL*Plus appears in Courier font. Structured Query Language (SQL) key words appear in upper case. Table and attribute names appear in lower case. "SQL" represents the SQL*Plus command prompt. "2" and "3" represent the SQL*Plus continuation prompt. <u>Do not</u> type them when they appear in these positions.
	
	In this document, commands are sometimes split to accommodate the document's format. You can omit the continuation and type the entire command on a single line.

```% telnet``` *String-Name*	Italicized text indicates an item, such as a file name, that you must supply.
``% mac_startL0R [0	1]```	Items in square brackets separated by "|" indicate a set of options. Type one (and only one) of the options exactly as it appears.

Warnings and Notes appear throughout this users guide. Warnings alert the user to potentially destructive or hazardous actions. For example,

```
┌─────────────────┐
│    WARNING      │
└─────────────────┘
```

Selecting "Commit" will overwrite the value in the database. The original value cannot be recovered.

Notes provide points of useful information. For example,

NOTE

When logging on to LPS strings from an X terminal, the desktop will be on the first LPS string to which you connected. Window manager menus will apply to that string.

2. Getting Started

2.1 Introduction

This chapter describes the fundamentals of operating the LPS software. It explains the steps required to

- Log on to the LPS.

- Set up an environment to run the LPS software.

- Start up and shut down the LPS software.

- Navigate the LPS GUI.

- Set up Status and Error Message displays.

- Browse the LPS Journal to view status and error history.

2.2 Logging On to the LPS

If you do not already have accounts on the LPS system hosts, contact the LPS system administrator. Be sure to tell the system administrator that your account must be an LPS operator account. The system administrator will provide your user name and initial password.

Once you have a user name and password, you can log on to the LPS. Normally, you will log on to each of the four LPS operational strings by logging onto two strings from each of the LPS X terminals. However, you can log onto any set of the strings from either of the X terminals or from the Indy workstations normally used for the MWD.

NOTE

> When logging on to LPS strings from an X terminal, the desktop will be on the first LPS string to which you connected. Window manager menus will apply to that string.

2.2.1 Logging On From an X Terminal

To log on to the first LPS string from an X terminal:

1. If necessary, press the "Setup" key to display the X terminal menu.

2. If necessary, select the **Login>Login New X Session…** menu option to display the X session login dialog.

3. Make sure the **Net** field is set to **TCP** and the **Type** field is set to **Host**.

4. Determine the IP address of the X terminal by

   – Selecting the **Setup>Change Setup Parameters…** menu option

–   Clicking on the box to the left of **IP** to display the Internet Parameters Setup form

–   Finding the Internet address of the X terminal in the "IP Address" field and making a note of it

–   Closing the "Change the Setup Parameters..." menu by clicking the left mouse button over the bar in the upper left-hand corner of the window to pop up a menu and then clicking the left mouse button over the "Close" option

–   Closing the "Setup" menu box by clicking the left mouse button over the bar in the upper left-hand corner of the window to pop up a menu and then clicking the left mouse button over the "Close" option

5. Connect to an LPS string by selecting its name from the **Default Hosts** list or by typing its name or IP address into the **Host field** (move the cursor into the typing area to enter the name) and then click **OK** with any mouse button.

6. A login dialog will appear after a few seconds. Type your user name and press return, then type your password (your password is not echoed on the screen) and press return.

A desktop on the LPS string will appear. To log on to additional LPS strings from an X terminal:

7. Select an existing xwsh or xterm window or create a new window.

8. In the selected window, type the following command:

    `% telnet ` *String-Name-or-IP-Address*

9. At the `login:` prompt, type your user name and press return; at the `Password` prompt type your password (your password is not echoed on the screen) and press return.

10. Set the DISPLAY environment variable by typing

    `% setenv DISPLAY ` *X-Terminal-IP-Address*`:0.0`

    X-Terminal-IP-Address is the IP address for the X terminal that you discovered in step 4.

Repeat steps 7 through 10 for each LPS string that you want to connect with from this X terminal.

2.2.2 Logging On From a Workstation

To log on to a LPS string from a workstation:

1. Log on to the workstation. Type your user name and press return, then type your password (your password is not echoed on the screen) and press return.

2. Select an existing window or create a new window.

3. In the selected window, type the following command:

    `% telnet ` *String-Name*

4. At the `login:` prompt, type your user name and press return; at the `Password` prompt, type your password (your password is not echoed on the screen) and press return.

Repeat steps 2 through 4 for each LPS string you want to connect with from this workstation.

2.3 Setting Up Your Environment

The LPS software includes a resource file that will set up your csh or tcsh environment properly [there is no support for Bourne (sh) shell or Korn (ksh) shell]. To invoke the file, add the following lines at the top of your .cshrc file on each LPS string:

```
% setenv LPS_HOME /user/LPS/st/r1 (this is an example)
% source $LPS_HOME/.lpsrc
```

NOTE

You have separate accounts on each LPS string and workstation and each uses a different .cshrc file. Lines similar to those above must appear in each of these files.

If necessary, you can modify any of the LPS elements. Appendix D describes how to do it.

2.4 User Permissions and Privileges

NOT IMPLEMENTED IN RELEASE 2

2.5 Starting Up the LPS Software

Once you have logged on to each LPS string, you will need to start up the LPS software on each string.

NOTE

A separate instance of the LPS software must be started on each LPS string.

To start the LPS software on a particular LPS string:

1. Move to the X terminal or workstation from which you are logged on to the string.

2. Use the mouse to move the cursor into a shell window for that string.

3. Set up to display LPS startup messages in the window by typing:

```
% set LPSJournal = \
? `grep local0.debug /etc/syslog.conf | awk '{print $NF}'`
% tail -f $LPSJournal&
```

4. Start up the LPS software by typing

```
% mac_lps
```

5. The LPS software will update raw data file accounting and LPS process information in the LPS database. Any error messages encountered will both display in your window and be written to the LPS Journal file.

6. If the database is updated successfully, the LPS GUI will start up.

7. Stop the display of LPS messages to the window by typing `% killall tail` in the window.

2.6 Navigating the LPS GUI

You control the LPS through the LPS software GUI. The LPS GUI displays as a menu bar (Figure 2–1).
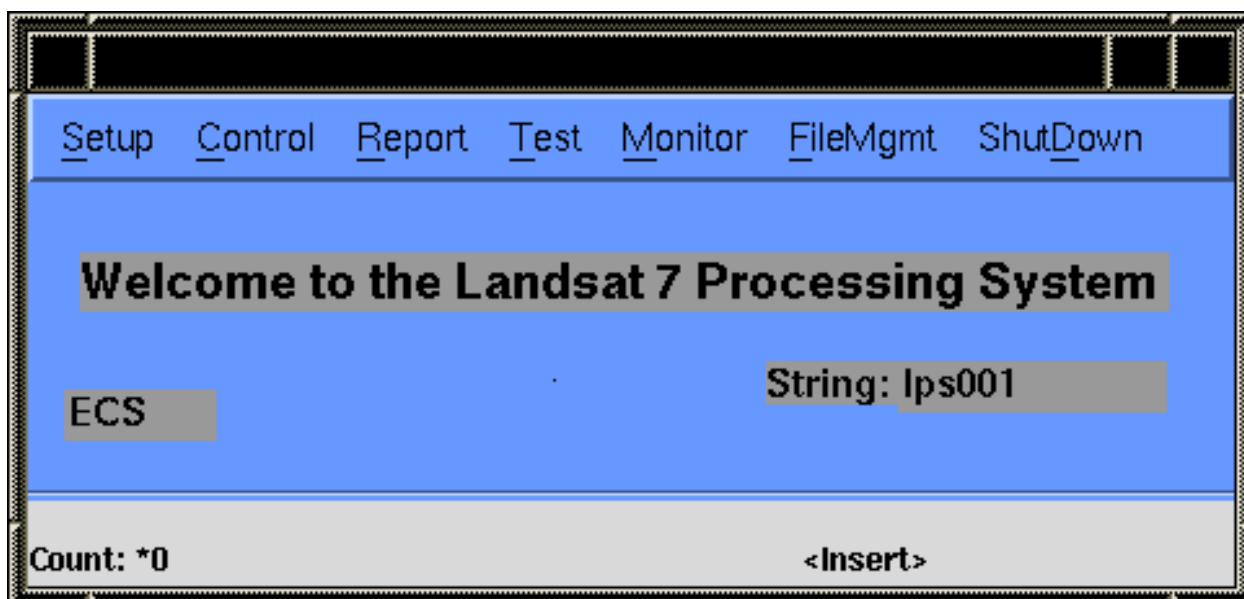
Figure 2–1. LPS GUI Menu Bar

NOTE

Each LPS string has a separate GUI. Commands issued through the GUI apply only to that string. The GUI displays the name of the string on which it is executing.

To issue a command through the LPS GUI on a particular LPS string:

1. Hold down the left mouse button over the desired menu item.

2. While still holding down the mouse button, move the cursor to highlight the selection you want to invoke.

3. Release the mouse button.

In most cases, a dialog form will appear asking you to confirm your selection and possibly asking for further information. Details for each command accessible through the LPS GUI and instructions for providing any further information appear in Section 10.

2.7 Status and Error Message Displays

All of the LPS software writes status and error messages to an LPS Journal file. You monitor LPS operations by reviewing the messages sent to the LPS Journal. To view messages as they are sent to the LPS Journal, set up a status/error message display window after you start up the LPS software on each string. You can also browse the LPS Journal file to review LPS processing history and add a message to the LPS Journal during processing.

NOTE

Each string has a separate LPS Journal file. Status and error messages appear only in the Journal file for the string on which they were generated.

2.7.1 Setting Up Message Displays

To set up a status/error message display:

1. Select the **MONITOR>Display Operations Messages...** menu option. The LPS GUI will display a dialog allowing you to select the priorities of the messages to be displayed.

2. If necessary, select priorities for display. The LPS software will display only messages with priorities that you have selected. The default is to display messages of all priorities except DEBUG.

3. Click the OK button; the LPS software will display a new window with 10 messages already written to the LPS Journal file.

NOTE

Any number of message displays can be created displaying messages of different priorities. For example, you could create one window displaying only high-priority error messages and another displaying informational messages. Because fewer messages go to the high-priority message window, it is less likely that an important message will scroll off the screen before you can see it.

The following is an example of a line from the LPS Journal file as displayed in the "Display Operations Messages..." window:

```
15:36:22 INFO syslog: 3475 [ldt_establish_client.c:270] ECS
                accepted authentication request.
```

The time of the message appears first, followed by its priority (INFO in the example) and source. Table 2–1 summarizes the priorities and their meanings. For messages generated by the LPS software, the source is always "syslog." The process group identifier (ID) of the process generating the message appears next (3475 in the example). The unit name and source line

appear next in square brackets. These are normally of interest only for software troubleshooting. The final part of the line is the message itself. Note that the message wraps to fit within the window.

Table 2–1. LPS Message Priorities

Priority No.	Description
EMERG	EMERGENCY – system is unusable
ALERT	ALERT – immediate action must be taken
CRIT	CRITICAL – critical condition
ERROR	ERROR – error condition
WARN	WARNING – warning condition
NOTICE	NOTICE – normal but significant
INFO	INFO – informational message
DEBUG	DEBUG – debug level messages intended for software trouble-shooting, not of operational interest

Normally, you will create a status/error message display window for each LPS string by selecting menu options from the LPS GUI. It is also possible to view LPS messages as they are written to the LPS Journal file from an IRIX shell.

2.7.2 Browsing the LPS Journal

To browse the LPS Journal:

1. Select the **MONITOR>Display LPS Journal File...** menu option. The LPS GUI will bring up the IRIX sysmon(1) utility to display the LPS Journal file (Figure 2–2).

2. If desired, customize the display by selecting options from the sysmon(1) menu bar. Consult the online documentation for sysmon(1) for help in customizing the display by clicking the "Help" option on the far right of the sysmon(1) window menu bar.

2.7.3 Adding a Message to the LPS Journal

To add a message to the LPS Journal during processing:

1. Select the **MONITOR>Add LPS Journal Entry...** menu option. The LPS GUI will display a box into which you can type your entry.
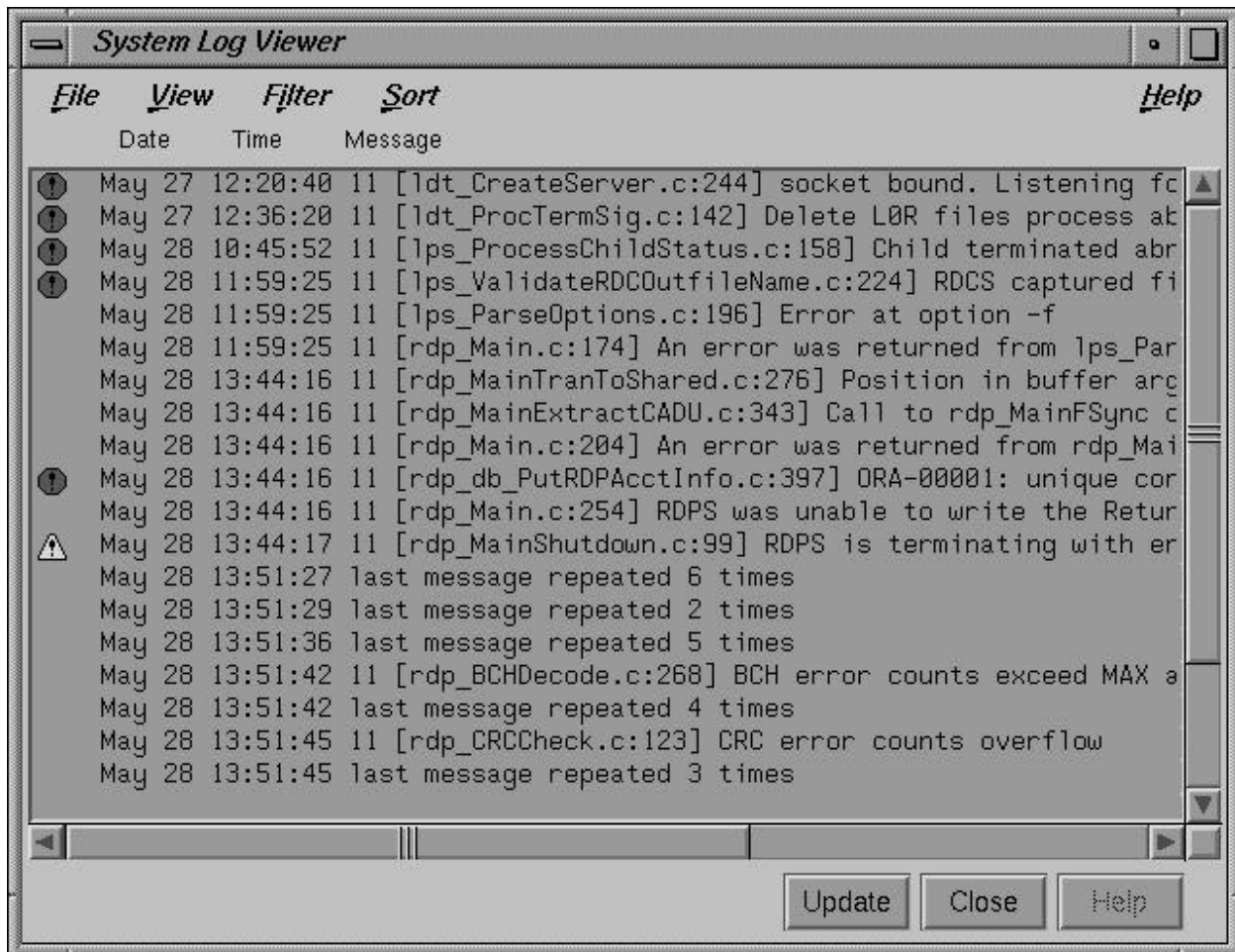
Figure 2–2. LPS Journal File Displayed by sysmon(1)

2. Type your entry in the box.

3. Click the OK button to send your entry to the LPS Journal file.

2.8 Shutting Down the LPS Software

To shut down the LPS software on a particular string:

1. The user does not need to verify that the DDN handler (rcvddn) is not currently processing a DDN. If this is desired anyway, determine whether the DDN handler is active by reviewing the LPS Journal file for DDN-related messages. The LPS shutdown process will wait until the current DDN is processed before shutting down.

2. Select the **Shutdown** menu option.

3. If the LPS software displays a dialog form stating that LPS processes are active and you still wish to shut down the LPS software:

   3a. Select the **Stop Data Capture...** option from the **Control** menu. Click the OK button when the confirmation dialog appears.

3b. Select the **Stop L0R Processing...** option from the **Control** menu. When the Stop Level 0R Processing... dialog appears, select an instance of L0R processing from the menu and click the OK button. Repeat this step until all Level 0R processes have been stopped.

3c. Select the **Shutdown** menu option.

4. Click the OK button in the confirmation dialog. Clicking the CANCEL button will leave the LPS software up and running.

2.9 Logging Off the LPS

Logging off from the LPS strings in an orderly manner ensures that all resources are freed and that all functions to be executed on exit are performed.

To log off from each LPS string to which you are connected by telnet:

1. Exit from all programs executing in separate windows, such as LPS Journal file displays using xwsh, windows-based editors, and sysmon.

2. Verify that the LPS GUI has been shut down and that no LPS software processes invoked from the command line are still running (see Section 2.8).

3. Either type `logout` in the telnet session window or select **Quit** from the window's right mouse button menu.

To log off from an LPS string to which you are directly logged on:

4. Select **Logout** from the desktop's right mouse button menu.

5. Click the **Yes** button in the confirmation dialog.

6. If you are logged on through an X terminal, click the **Yes** button in the X terminal's confirmation dialog.

3. Data Capture Operations

3.1 Introduction

This chapter describes how to perform operations related to LPS data capture. It describes how to

- Manage LPS automatic data capture operations.

- Capture data manually.

- Monitor data capture.

3.2 Managing Automatic Data Capture

This section describes how to perform operations related to LPS automatic data capture. It describes how to

- Fetch a contact schedule from the Indy workstation.

- View the contact schedule file.

- Ingest the contact schedule file into a string database.

- View or edit the contact schedule stored in a string database.

- View or edit a string's capture source.

- Verify whether automatic data capture is active.

Note that the station of origin is automatically set to EDC for automatic data capture.

3.2.1 Retrieving a Contact Schedule From the Indy Workstation

The MOC will place the contact schedule on an Indy workstation. Use IRIX FTP to retrieve it to the development string. Use the get command. The file will have the naming convention L7yyyydddLPSSCH.Snn, where

L7	is the mission ID
yyyy	is the year for which the schedule applies
ddd	is the first day of the interval for which the schedule applies (001-366)
LPSSCH	indicates that the file contains the contact schedule for the LPS
S	specifies the sequence number follows
nn	is the sequence number of the file for the year and day of year (00-99)

3.2.2 Viewing the Contact Schedule File

Use your favorite UNIX text viewing command to examine the file as received. Examples are cat, vi, emacs, or more.

3.2.3 Ingesting a Contact Schedule File

To ingest a contact schedule file:

1. Select the **Setup>Ingest Contact Schedules** option from the LPS GUI.

2. Select the contact schedule file of interest, highlighting it with a mouse touch.

3. Select the strings to propagate the file to by clicking in their respective boxes.

4. Click the OK button.

3.2.4 Viewing or Editing a String's Contact Schedule

To view or edit a string's contact schedule:

1. Select the **Setup>View/Edit Contact Schedule** option from the LPS GUI.

2. This displays the scheduled start and stop times of the data in DD-MMM-YY-HH:MM:SS format. Scroll down to see more times if necessary.

3. Edit an entry by highlighting the item in question with the mouse and then typing in the correction as needed.

4. Click the OK button to accept the changes or the CANCEL button to reject the changes. ORACLE forms gives back a confirmation on use of the OK button and checks the start and stop times for format and logical errors.

5. Click the INSERT button to insert a new start and stop time.

6. Delete an entry pair by highlighting and then clicking the DELETE button. A delete confirmation window will appear to ask for confirmation of the deletion. Hit OK to confirm. The entry will no longer appear in the window, but until the final OK button is clicked, the database has not been affected.

7. Click the CANCEL button to leave the window without changes.

8. Click the OK button to leave the window with changes.

3.2.5 Viewing a String's Capture Source

Select the **Setup>View/Edit Capture Source** option from the LPS GUI. The current capture source will appear.

3.2.6 Editing a String's Capture Source

To edit a string's capture source:

1. Select the **Setup>View/Edit Capture Source** option from the LPS GUI.

2. Pull down the list of capture sources and select one.

3. Click the OK button to set and exit.

```
WARNING
```

The LPS software uses the capture source to determine which entries in the contact schedule apply to the string. An incorrect capture source value may result in the failure to capture data on schedule.

The LPS software also uses the capture source to identify the data in its accounting. An incorrect capture source value may result in misidentification of captured data.

3.2.7 Verifying That Automatic Data Capture Is Active

Select the **Control>Stop Auto Capture** option. The **Stop Auto Capture** option is available when automatic data capture is active. Select the **Control>Start Auto Capture** option. The **Start Auto Capture** option is available when automated data capture is not active.

3.3 Manual Data Capture

The LPS software captures most data automatically according to the contact schedule you receive from the MOC. Data from supplementary ground stations such as Alaska (AGS) and Svalbard (SGS) will arrive on tape to the LGS and must be captured manually by the LPS. Also, for testing or for handling emergency situations, you may need to start and stop data capture yourself.

```
WARNING
```

Data capture is independent on each LPS string. To capture all data channels, you must command each LPS string to capture data.

You will normally capture data manually through the LPS GUI. It is possible, though less convenient, to do so directly from an IRIX shell. The shell interface is intended for emergency situations in which capture must occur when the LPS GUI is not up and running and for use in shell scripts you may create to simplify manual operations. For details see Appendix E, Section E.3.1. Root permission is needed to capture data.

3.3.1 Starting Data Capture

To start data capture manually on an LPS string:

1. Select **Control>Start Capture...** from the LPS GUI. This menu option is available only if the LPS software is not currently capturing raw wideband data.

2. The LPS software displays a Start Data Capture dialog (Figure 10–12).

3. If necessary, set the capture source to the source of the data to be captured by holding down the mouse button in the capture source field and selecting the appropriate value from the pop-up menu. See Table 10–1 for a description of the capture source values.

4. If necessary, set the capture duration to how long capture should continue from the point of first data capture. The format of the time is min:sec. The default time is 14 minutes, which is the longest capture time for a single contact.

5. If necessary, change the station of origin from the default of EDC to one of the other stations (i.e., AGS or SGS) by holding down the mouse button in the station of origin field and selecting the appropriate value.

6. If necessary, set the capture process execution characteristics controls. The capture processor may be isolated to devote itself for the capture. Also, L0R processing may be suspended on the string.

7. Click the OK button.

8. Click the YES button in the confirmation dialog.

Data capture begins as soon as you click the YES button, though there is a small delay as the LPS software sets up to begin capture.

3.3.2 Stopping Data Capture

To stop data capture manually on an LPS string:

1. Select **Control>Stop Capture...** from the LPS GUI. This menu option is available only if the LPS software is currently capturing raw wideband data.

2. The LPS software displays a Stop Data Capture dialog (Figure 10–13) asking you to confirm your command.

3. Click the OK button.

Data capture stops as soon as you click the OK button.

3.4 Monitoring Data Capture

To monitor data capture operations, view the status and error messages generated by the mac_AutoCapture and rdc processes. To view status and error messages generated by these processes, set up a real-time display of the LPS Journal file (see Section 2.7.2).

4. Raw Wideband Data Management

4.1 Introduction

This chapter describes how to perform operations related to LPS raw wideband data management. It describes how to

- Copy raw data to tape

- Stop copying raw data to tape

- Restore raw data from tape to disk

- Stop restoring raw data from tape to disk

- Generate a tape label manually

- Delete raw data files

Nominally, data is automatically archived during the auto capture process. This chapter is for manual operations.

4.2 Copy Raw Data to Tape

To copy raw data to tape:

1. Select **Control>Start Copy To Tape** from the LPS GUI.

2. Select which raw data file to copy to tape highlighting the line chosen.

3. Click the OK button.

The tape copy goes on in background. Only one raw data file can be copied to tape at a time.

4.3 Stop Copy to Tape

To stop copying to tape:

1. Select **Control>Stop Copy To Tape** from the LPS GUI. This window tells you which file is currently being processed.

2. Click the OK button.

3. A confirmation window will quiz you on your decision. Click the OK button if you wish to proceed with this course of action.

4.4 Restore Raw Data From Tape to Disk (Restage)

To restore raw data from tape to disk:

1. Select **Control>Start Restage** from the LPS GUI. This will extract raw data from the tape in slot 6. The user must either manually put the tape in the slot or submit the command to load the tape first.

2. Click the OK button.

4.5 Stop Restage

To stop restage:

1. Select **Control>Stop Restage** from the LPS GUI.

2. Click the OK button.

3. A confirmation window will verify your decision. Click the OK button if you wish to proceed with this course of action.

4.6 Generate Tape Label

Normally, a tape label is automatically generated when copying raw data to tape. However, in the event of some mishap, a tape label can be manually generated.

1. Select **Control>Generate Tape Label** from the LPS GUI.

2. Select the raw date file of interest by highlighting the row with the mouse.

3. Click the OK button.

4.7 Delete Raw Data Files

In automatic processing, raw data files are deleted after being processed and archived. The Start L0R Processing GUI also has a selection to delete raw data files after processing. If raw data files have not been deleted, they may still be deleted through the GUI. Root permission is needed to delete files.

To delete raw data files:

1. Select **Filemgmt>Delete Raw File** from the LPS GUI.

2. Select a raw data file. The volume in megabytes and whether it has been archived and processed will appear on the screen.

3. Click the OK button to act on the decision.

5. Level 0R Processing Management

5.1 Introduction

This chapter describes how to perform operations related to LPS L0R processing. It describes how to

- Manage automatic L0R processing operations.

- Perform L0R processing manually through the LPS GUI.

- Monitor L0R processing.

- View and edit L0R processing parameters.

- View and edit L0R processing error reporting thresholds.

5.2 Automatic L0R Processing Management

This section describes how to start and stop automatic control of LPS L0R processing. This action is done through the automatic data capture procedure.

To start automatic data capture:

1. Select the **Control>Start Auto Capture** command from the LPS GUI.

2. Click the OK button.

Clicking the OK button will automatically capture data at the designated time in the schedule, automatically perform L0R processing to the data, and automatically archive raw data.

To stop automatic data capture:

1. Select the **Control>Stop Auto Capture** command from the LPS GUI.

2. Click the OK button.

Clicking the OK button will kill all the child processes that had been generated by the automatic data capture process.

5.3 Manual L0R Processing

This section describes how to manually control LPS L0R processing. You must start L0R processing manually under the following conditions:

- To process a raw data file for which automatic L0R processing failed to complete

- To handle a reprocessing request

- To perform tests

You can stop any L0R processing, whether it was started automatically or manually.

You will normally control L0R processing manually through the LPS GUI. It is also possible to control L0R processing directly from an IRIX shell (see Appendix E). The IRIX interface is intended for use in shell scripts you may create.

NOTE

When you stop L0R processing manually, all output files created by the processing will be deleted.

5.3.1 Starting L0R Processing

To manually start L0R processing:

1. Select the **Control>Start L0R Processing...** command from the LPS GUI. The "Start Level 0R Processing" dialog appears.

2. Select the raw wideband data file you want to process by clicking on any of its information fields. Because entries are ordered in descending order by contact sequence ID, newer entries appear at the top.

3. The file version number field can be edited. If this raw data has already been processed with this file number on another string, it should be incremented.

4. If desired, change the settings for the options to delete the raw data file after successful processing (not selected by default) or to delete the output files automatically when transferred (selected by default).

5. Click the OK button. A confirmation dialog appears.

6. Click the OK button.

5.3.2 Stopping L0R Processing

To manually stop L0R processing:

1. Select the **Control>Stop L0R Processing...** command from the LPS GUI. The "Stop Level 0R Processing" dialog appears.

2. Select the raw data file and instance of L0R processing to terminate by clicking on any of its information fields.

3. Click the OK button. A confirmation dialog appears.

4. Click the OK button.

5.4 Monitoring L0R Processing

L0R processing can be monitored in two ways:

1. *Viewing L0R messages* – A status message indicates that L0R processing has completed. L0R error messages indicate when the amount of erroneous data

encountered exceeds the user-specified threshold. Error messages also indicate (generally fatal) processing errors encountered by the LPS L0R processing software.

2. *Using the MWD* – The MWD provides a color thumbnail image of the band data as it is being processed. The image gives a rough idea of processing quality.

5.4.1 Viewing L0R Status and Error Messages

To view status and error messages generated by the LPS L0R processing software, set up a real-time display of the LPS Journal file by any of the methods described in Section 2.7.2. The LPS software writes all status and error messages to the LPS Journal file (and to the standard output, if the LPS_LOG_STDOUT environment variable is set).

5.4.2 Moving Window Display

The MWD is an X11/Motif graphics application that comes up automatically as part of the image data processing subsystem (IDPS) during LPS processing. It runs concurrently with band file generation. It is a display window that contains a color representation of the swath as scan data is being processed by the IDPS. The user can select from the LPS GUI which three of the bands are to be displayed and the RGB colors that are to represent them in the image. At the bottom of the display, a set of labels indicate the subinterval sequence ID, the format (1 or 2), the time of the current scan, and the band-color associations. The MWD terminates automatically when the last scan in the contact period has been processed.

The form and location of the MWD can be configured by entering X11/Motif parameters in the XOPTIONS field of the VALID_MWD_PARMS table through the GUI.

The Oracle Forms GUI and the MWD do not like each other. They are both X11/Motif applications, and they fight over control of the X Windows color palette if the MWD is being viewed on the same terminal. If MWD is being displayed on a separate terminal (the nominal configuration), there is no need for this action. Fortunately, this problem can be avoided.

Oracle Forms has a resource file: $ORACLE_HOME/guicommon/tk2/admin/Tk2Motif.gray. The LPS user needs to edit this file (or a copy of it in $HOME) and set the "Tk2Motif*UsePrivateColormap: True" parameter. This causes Forms to use a private Motif color map so that it does not "gobble up" the system color map and trip up the MWD. NOTE: Background forms will turn black if this is done. Forms in the foreground should remain unaffected.

5.5 L0R Processing Parameters Management

This section describes how to view and modify the parameters that the LPS software uses during L0R processing. L0R processing parameters are values that control certain aspects of L0R processing, such as the number of bit errors to allow in a frame synchronization marker or the maximum forward time jump length allowed within a subinterval. The L0R processing parameters fall into the following categories:

- *Raw Data Processing* – Parameters that control Consultative Committee for Space Data Systems (CCSDS) frame synchronization and error detection and correction.

- *Major Frame Processing* – Parameters that control the reconstruction of ETM+ telemetry data and the detection of subintervals within the contact.

- *PCD Processing* – Parameters that control the reconstruction of PCD telemetry data and the identification of WRS scenes.

- *Band Processing* – Parameters that control browse file generation and the Automated Cloud Cover Assessment (ACCA).

- *Moving Window Display Processing* – Parameters that control the construction of the MWD.

- *LPS Configuration Metadata* – Parameters that identify and characterize certain overall aspects of L0R processing.

Table 5–1 lists the L0R processing parameters that are in the system database, their table name, and their attribute name. Table 5–2 lists the suggested values and valid ranges of these L0R parameters obtained from the IAS calibration parameter file (CPF). Table 5–3 lists the suggested values and valid ranges of the remaining L0R parameters in the system database. L0R parameters relating to LPS/EDC DAAC communications are discussed in Section 6.2.

Table 5–1. Table and Attribute Names for L0R Processing Parameters (1 of 7)

Parameter	Table Name	Attribute Name(s)
Band Processing – Browse		
Browse Bands Three bands to include in browse image representing red, green, and blue, respectively NOTE: Bands must be included in format 1.	valid_band_parms	multi1, multi2, multi3
Browse Subsampling Ratio Browse image subsampling reduction ratio	valid_band_parms	subs
Browse Wavelets Iterations Number of times to apply wavelets reduction during browse generation	valid_band_parms	wave
JPEG Quality Factor Compression quality factor for reducing browse images	valid_band_parms	JPEG_Quality
Contrast Stretch Factor Clipping factor used in clipping ends of multi-band browse contrast values	valid_band_parms	contrast_stretch_ factor

Parameter	Table Name	Attribute Name(s)
Radiometric Calibration Gains Gains used for ACCA and browse image radiometric correction; there are separate values for high and low gains for each detector in each format 1 band (1–6) for a total of 176 values. Updated through the IAS CPF. Not on GUI.	valid_detector_ gain_bias	high_gain, low_gain
Radiometric Calibration Biases Biases used for ACCA and browse image radiometric correction; there are separate values for high and low gains for each detector in each format 1 band (1–6) for a total of 176 values. Updated through the IAS CPF. Not on GUI.	valid_detector_ gain_bias	high_gain_bias, low_gain_bias
Raw Data Processing – CCSDS Frame Synchronization		
Bit Slip Correction Extent Number of bits around expected frame synchronization pattern start bit to search for pattern	valid_ccsds_parms	cadu_bit_slip_ correction_extent
Check Synchronization Marker Error Tolerance Number of bit errors in frame synchronization pattern to accept and remain in Check mode	valid_ccsds_parms	cadu_sync_mark_ check_error_tol
Check Tolerance Number of channel access data unit (CADU) frame synchronization errors to tolerate while in Check mode before going back to Search mode	valid_ccsds_parms	cadu_check_tol
Flywheel Tolerance Number of CADU frame synchronization errors to tolerate while in Flywheel mode before going back to Search mode	valid_ccsds_parms	cadu_flywheel_tol
Lock Synchronization Marker Error Tolerance Number of bit errors in frame synchronization pattern to accept and remain in Lock mode	valid_ccsds_parms	cadu_sync_lock_ error_tol
Transfer Frame Trouble Flag Flag to generate transfer from trouble file	valid_ccsds_parms	xfer_frame_trouble_ file

Parameter	Table Name	Attribute Name(s)
Bose-Chaudhuri-Hocquenghem (BCH) Error Checking Flag Flag to perform BCH error checking	valid_ccsds_parms	BCH_flag
Search Tolerance Number of successful CADU frame synchronizations required before moving from Search to Check mode	valid_ccsds_parms	cadu_search_tol
CCSDS Parameters ID Valid_CCSDS_Parm stable's record ID (internal, not in GUI)	valid_ccsds_parms	CCSDS_parms_id
Insertion Time Insertion time of record (internal, not in GUI)	valid_ccsds_parms	insertion_time
Moving Window Display Processing		
Format 1 Red Band Band chosen to be red band for MWD for format 1 data	valid_MWD_parms	Fmt1_Red_Band
Format 1 Green Band Band chosen to be green band for MWD for format 1 data	valid_MWD_parms	Fmt1_Green_Band
Format 1 Blue Band Band chosen to be blue band for MWD for format 1 data	valid_MWD_parms	Fmt1_Blue_Band
Format 2 Red Band Band chosen to be red band for MWD for format 2 data	valid_MWD_parms	Fmt2_Red_Band
Format 2 Green Band Band chosen to be green band for MWD for format 2 data	valid_MWD_parms	Fmt2_Green_Band
Format 2 Blue Band Band chosen to be blue band for MWD for format 2 data	valid_MWD_parms	Fmt2_Blue_Band

Table 5–1. Table and Attribute Names for L0R Processing Parameters (4 of 7)

Parameter	Table Name	Attribute Name(s)
Xoptions Options for X Window display of moving window	valid_MWD_parms	Xoptions (see note at end of Section 5.5.1)
Major Frame Processing		
ETM+ Major Frame Fill Pattern Two-byte pattern to use as fill for missing ETM+ telemetry	valid_mfp_parms	fill_value
ETM+ Majority Vote Tolerance Tolerable number of differing bit values in any 40-bit majority vote	valid_mfp_parms	maj_vote_tol
Maximum Timespan Expected timespan of major frame over a certain number of days window	valid_mfp_parms	max_time_span
End-of-Line (EOL) Tolerance Bilateral search zone around nominal location for EOL marker in number minor frames	valid_mfp_parms	eol_tol
Major Frame Synchronization Tolerance Bilateral search zone around nominal location for synchronization marker in number minor frames	valid_mfp_parms	mjf_sync_tol
Major Frame Synchronization Size Minimum number of bytes needed in synchronization pattern	valid_mfp_parms	mjf_sync_size
Maximum Minor Frame Counter Maximum the minor frame counter should run to	valid_mfp_parms	max_mnf_counter
Minimum Major Frame Count Minimum number of scans (major frames) in a subinterval to keep output files (for Release 2 on thresholds GUI under subinterval size)	valid_mfp_parms	min_mjf_count_per_sub_intv

Table 5–1. Table and Attribute Names for L0R Processing Parameters (5 of 7)

Parameter	Table Name	Attribute Name(s)
ETM+ Trouble File Flag Flag to generate a trouble file for ETM+ processing (not effective in Release 2)	valid_mfp_parms	ETM_plus_trouble_ file
Nominal Major Frame Data Period Nominal major frame transmission rate in seconds	valid_mfp_parms	mjf_data_period
Sensor Alignments Adjustment in instantaneous field of view (IFOV) to layout geometry and multiplexer sampling; there are separate values for forward and reverse scans for odd and even detectors for each band for a total of 32 values (forward odd, forward even, reverse odd, and reverse even for Bands 1 through 8) (NOTE: Manual update is slightly different than other parameters. See Section E.5.3.3.)	valid_sensor_align_ parms	X_Y_BZ X = Forward or Reverse Y = Even or Odd Z = 1..8 e.g., Forward_ Even_B1
Subinterval Delta Time Maximum time gap, in seconds, between ETM+ major frame times that will be filled; time gaps greater than this value will cause telemetry to be placed in a new subinterval	valid_mfp_parms	sub_intv_delta
Time Range Tolerance Tolerable difference between actual ETM+ major frame timecode value and expected value	valid_mfp_parms	time_range_tol
LPS Configuration Metadata		
Capture Source Character string used to identity the capture source (not on GUI)	lps_configuration	capture_source
Instrument ID ETM+	lps_configuration	Instrument ID
IAS Parameter File Name Name of IAS parameter file used by LPS string	lps_configuration	IAS_PARAM_ FILE_NAME

Table 5–1. Table and Attribute Names for L0R Processing Parameters (6 of 7)

Parameter	Table Name	Attribute Name(s)
LPS Software Version Number Version number of LPS software installed on LPS string (for format, see entry in Table 5–3)	lps_configuration	lps_software_ version_number
LPS String ID Character string used to identify LPS string (fixed on GUI)	lps_configuration	lps_hardware_ string_id
Spacecraft Identifier (SCID) Character string used as SCID in metadata files	lps_configuration	spacecraft_id
PCD Processing		
PCD Fill Pattern Byte value used as fill for missing PCD telemetry for internal processing (This parameter if now obsolete, but some routines still access the table.)	valid_pcd_parms	pcd_frame_fill
PCD Processing – WRS Scene Identification		
Attitude Quaternion Tolerance Maximum tolerable absolute value of difference between sum of squares of attitude quaternion and 1.0	valid_pcd_thres	attitude_ quaternion_tol
Earth Semi-Major Axis Earth's semi-major axis in meters from center of Earth to the equator	valid_scene_parms	semi_major_axis
Earth Semi-Minor Axis Earth's semi-minor axis in meters from center of Earth to the poles	valid_scene_parms	semi_minor_axis
Ephemeris Position Lower Bound Lower bound on valid ephemeris position vector magnitude	valid_pcd_thres	ephem_position_ lower
Ephemeris Position Upper Bound Upper bound on valid ephemeris position	valid_pcd_thres	ephem_position_ upper

Parameter	Table Name	Attribute Name(s)
Ephemeris Velocity Lower Bound Lower bound on valid ephemeris velocity vector magnitude	valid_pcd_thres	ephem_velocity_lower
Ephemeris Velocity Upper Bound Upper bound on valid ephemeris velocity vector magnitude	valid_pcd_thres	ephem_velocity_upper
Ephemeris Cross Product Maximum Largest valid cross-product of ephemeris velocity and position	valid_pcd_thres	ephem_cross product_max
Ephemeris Cross Product Minimum Smallest valid cross-product of ephemeris velocity and position	valid_pcd_thres	ephem_cross product_min
ETM+ to Attitude Control Reference Axis Matrix Matrix describing relationship of ETM+ optical axis to satellite's attitude control reference axis; each element of matrix is stored as a separate field in LPS database	valid_scene_parms	etm_body_transmatrix_XY X = 1..3 Y = 1..3
UT1 to Universal Time Coordinated (UTC) Time Conversion Coefficients UT1-UTC time difference to be used for UT1 to UTC time conversions for each day of a 180-day period. Each value is stored as a separate field in the database, tagged with the day on which it is valid. (NOTE: Manual update is slightly different from other parameters. See Section E.5.3.3.)	UTC_UT1_corrections	UT1_corrections

Table 5–2. IAS L0R Parameters Suggested Values and Valid Ranges

Parameter	Suggested Value	Valid Range	Description
Band Processing – Browse			
Radiometric Calibration Gains	TBD post-launch	TBD post-launch	Gains used for ACCA and browse image radiometric correction; there are separate values for high and low gains for each detector in each format 1 band (1–6) for a total of 176 values. Updated through the IAS CPF. Not on GUI.
Radiometric Calibration Biases	TBD post-launch	TBD post-launch	Biases used for ACCA and browse image radiometric correction; there are separate values for high and low gains for each detector in each format 1 band (1–6) for a total of 176 values. Updated through the IAS CPF. Not on GUI.
Major Frame Processing			
Sensor Alignments	TBS by Science Office	0-287	Adjustment in IFOVs to layout geometry and multiplexer sampling; there are separate values for forward and reverse scans for odd and even detectors for each band for a total of 32 values (forward odd, forward even, reverse odd, and reverse even for Bands 1 through 8)
LPS Configuration Metadata			
IAS Parameter File Name	N/A	N/A	Name of IAS parameter file used by LPS string in L7CPFyyyymmdd_yyymmdd.nn format
PCD Processing – WRS Scene Identification			
Earth Semi-Major Axis	6378137.000		Earth's semi-major axis in meters from center of the Earth to equator
Earth Semi-Minor Axis	6356752.314		Earth' semi-minor axis in meters from center of the Earth to the poles
ETM+ to Attitude Control Reference Axis Matrix	TBD post-launch	-1.00–+1.00	Matrix describing relationship of ETM+ optical axis to satellite's attitude control reference axis; each element of matrix is stored as a separate field in LPS database
UT1 to UTC Time Coefficients	N/A	0-1	UT1-UTC time difference to be used for UT1 to UTC time conversions for each day of a 180-day period; each value is stored as a separate row in the database tagged with day on which it is valid (NOTE: Manual update is slightly different from other parameters. See Section E.5.3.3.)

Table 5–3. Non-IAS L0R Parameters Suggested Values and Valid Ranges (1 of 5)

Parameter	Suggested Value	Valid Range	Description
Browse			
Browse Bands	any 3	three of 1, 2, 3, 4, 5	Three bands to include in browse image representing red, green, and blue, respectively NOTE: Bands must be included in format 1.
Browse Subsampling Ratio	1	1	Browse image subsampling reduction ratio
Browse Wavelets Iterations	3	3	Number of times to apply wavelets reduction during browse generation
JPEG Quality Factor	90	0...100	Compression quality factor for reducing browse images
Contrast Stretch Factor	3	0...49	Clipping factor used in clipping ends of multi-band browse contrast values
Moving Window Display			
Format 1 Red Band	any valid	1–6	Band chosen to be red band for MWD for format 1 data
Format 1 Green Band	any valid	1–6	Band chosen to be green band for MWD for format 1 data
Format 1 Blue Band	any valid	1–6	Band chosen to be blue band for MWD for format 1 data
Format 2 Red Band	any valid	6–8	Band chosen to be red band for MWD for format 2 data
Format 2 Green Band	any valid	6–8	Band chosen to be green band for MWD for format 2 data
Format 2 Blue Band	any valid	6–8	Band chosen to be blue band for MWD for format 2 data
Xoptions	no options	character string	Options for X Window display of moving window (see note at end of Section 5.5.1)

Table 5–3. Non-IAS L0R Parameters Suggested Values and Valid Ranges (2 of 5)

Parameter	Suggested Value	Valid Range	Description
Raw Data Processing – CCSDS Frame Synchronization			
Bit Slip Correction Extent	0	0…3	Number of bits around expected frame synchronization pattern start bit to search for the pattern
Check Synchronization Marker Error Tolerance	0	0…3	Number of bit errors in frame synchronization pattern to accept and remain in Check mode
Check Tolerance	0	0…3	Number of CADU frame synchronization errors to tolerate while in Check mode before going back to Search mode
Flywheel Tolerance	0	0…3	Number of CADU frame synchronization errors to tolerate while in Flywheel mode before going back to Search mode
CCSDS Parms ID	1	> 0	Valid_CCSDS_Parm stable's record ID
Insertion Time	N/A	date	Insertion time of record
Transfer Frame Trouble Flag	1	0 = yes, 1 = no	Flag to generate transfer from trouble file
BCH Error Checking Flag	1	0, 1	Flag to perform BCH error checking; 0 = on erroneous virtual channel data unit (VCDU), 1 = on every VCDU
Lock Synchronization Marker Error Tolerance	0	0…3	Number of bit errors in frame synchronization pattern to accept and remain in Lock mode
Search Tolerance	1	1…3	Number of successful CADU frame synchronizations required before moving from Search to Check mode
Major Frame Processing			
ETM+ Major Frame Fill Pattern	255	255	Byte pattern to use as fill for missing ETM+ telemetry
ETM+ Majority Vote Tolerance	36	0–40	Tolerable number of differing bit values in any 40-bit majority vote

Table 5–3. Non-IAS L0R Parameters Suggested Values and Valid Ranges (3 of 5)

Parameter	Suggested Value	Valid Range	Description
Maximum Timespan	2	> 0	Expected timespan of major frame over a certain number of days window
EOL Tolerance	20	>= 0	Bilateral search zone around nominal location for EOL marker in number minor frames
Major Frame Synchronization Tolerance	20	>= 0	Bilateral search zone around nominal location for synchronization marker in number minor frames
Major Frame Synchronization Size	60	0 <= x <= 80	Minimum number of bytes needed in a synchronization pattern
Maximum Minor Frame Counter	7600	>= 0	Maximum the minor frame counter should run to NOTE: This value should be large enough to account for bumper wear as the mission progresses.
Minimum Major Frame Count	5	>= 0	Minimum number of scans (major frames) in a subinterval to keep output files
ETM+ Trouble File Flag	0	1 = yes, 0 = no	Flag to generate a trouble file for ETM+ processing
Nominal Major Frame Data Period	0.071375	> 0	Nominal major frame transmission rate in seconds
Subinterval Delta Time	23	> 0	Maximum time gap, in seconds, between ETM+ major frame times that will be filled; time gaps greater than this value will cause telemetry to be placed in a new subinterval
Time Range Tolerance	0.0005	0.00– 9999999.99	Tolerable difference between the actual ETM+ major frame timecode value and the expected value
LPS Configuration Metadata			
Capture Source	see Table 10–1	character string	Character string used to identity capture source
Instrument ID	ETM+	character string	Character string used as instrument ID in metadata files

Table 5–3. Non-IAS L0R Parameters Suggested Values and Valid Ranges (4 of 5)

Parameter	Suggested Value	Valid Range	Description
LPS Software Version Number	not applicable	X.Y.Z X = major release no. Y = minor release no. Z = patch (or engineering) release no. X, Y, and Z are all numeric values	Version number of LPS software installed on LPS string
LPS String ID	must be identical to IRIX host name	must be identical to IRIX host name	Character string used to identify LPS string
SCID	Landsat 7	character string	Character string used as SCID in metadata files
PCD Processing – WRS Scene Identification			
Attitude Quaternion Tolerance	0.01	0.00–9999.99	Maximum tolerable absolute value of difference between sum of squares of attitude quaternion and 1.0
Ephemeris Position Lower Bound	7050	-8.3886×10^3 x 8.3886×10^3	Lower bound on valid ephemeris position vector magnitude (kilometers)
Ephemeris Position Upper Bound	7120	-8.3886×10^3 x 8.3886×10^3	Upper bound on valid ephemeris position vector magnitude (kilometers)
Ephemeris Velocity Lower Bound	7.4	-8.0 km/s x 8.0 km/s	Lower bound on valid ephemeris velocity vector magnitude (kilometers per second)
Ephemeris Velocity Upper Bound	7.6	-8.0 km/s x 8.0 km/s	Upper bound on valid ephemeris velocity vector magnitude (kilometers per second)

Table 5–3. Non-IAS L0R Parameters Suggested Values and Valid Ranges (5 of 5)

Parameter	Suggested Value	Valid Range	Description
Ephemeris Cross-Product Maximum	53200	not null	Largest valid cross-product of ephemeris velocity and position (kilometers squared per second)
Ephemeris Cross-Product Minimum	53000	not null	Smallest valid cross-product of ephemeris velocity and position (kilometers squared per second)

5.5.1 Viewing or Editing L0R Parameters

This subsection describes how to view or edit the current values of the L0R processing parameters on a LPS string through the LPS GUI.

L0R parameters may be viewed on individual strings. They can be edited on one string. Changes may be propagated to the other strings from the Test instance. Propagation is described in Section 5.5.3. To view or edit L0R parameters:

1. Select the **Setup>View/Edit L0R Parameters** command from the LPS GUI. The string currently in use will appear on the display along with the parameters.

2. The screen itself is scrollable, as are certain areas of the screen. Scroll to the parameters in question.

3. To edit parameters, insert the desired value in the field.

    If values are out of the permitted range in the database, you will not be allowed to move to a different field. A message at the bottom of the screen will notify you of the valid range of the field in question.

4. Click the OK button to keep changes or click the CANCEL button to leave parameters unchanged.

5. If any L0R processes are registered in the database, a dialog window will be output to the screen, informing the operator that the database cannot be updated at the moment. The operator may then cancel out or press the OK button again, in case the L0R processes have ceased.

NOTE

The contents of the XOPTIONS field in the VALID_MWD_PARMS table is passed to the X11 system. It is not interpreted by the LPS software. A user who wishes to pass X11 configuration parameters to the X server running the MWD should place them in this field. For example, to display the MWD on a different X terminal, enter "–display <IPaddr>" in the

XOPTIONS field. See an X11/Motif manual for a description of the valid parameters.

5.5.2 Ingesting IAS Calibration Parameter Files

The LPS regularly receives the CPF from the IAS through FTP transfer. To load a CPF into the system:

1. Select the **Setup>Load IAS Parameter File** command from the LPS GUI.

2. From the list of available files, click on the desired one.

3. Click the OK button.

The L0R parameters associated with the CPF will be updated on the string in use. The parameters can be propagated to other strings along with other L0R parameters from the Test instance, as described in Section 5.5.3.

5.5.3 Propagating Parameters to Other Strings

To propagate both the IAS CPF and the other L0R parameters to other strings:

1. Select the **Setup>Propagate L0R Parameters** command from the LPS GUI.

2. Select the strings to which the parameters need to be propagated.

3. Click the OK button.

Only the backup string's Test instance parameters can be propagated to the selected strings. The parameter propagation is prohibited when L0R processing is in progress in any of the target strings.

5.5.4 Image Data Processing Radiometric Correction Parameters File

Several parameters are used by the Browse and ACCA image data processing processes. Some have already been specified in Tables 5–1 and 5–2 in the Band Processing category. Those parameters already exist in the database. There are some other parameters which in Release 2 are only in a flat file and not in the database. The values for these parameters are obtained by looking at the IAS CPF and manually inserting them in the file.

The file is radCorrect.parms and can be found in the $LPS_HOME/tables directory.

Following is a description of the file:

PCDR = post-calibration dynamic range
CTV = cloud threshold value
TEP = thermal evaluation parameter

PCDR B1 maximum low radiance value   PCDR B1 maximum high radiance value
PCDR B1 minimum low radiance value   PCDR B1 minimum high radiance value
PCDR B1 bandwidth
PCDR B2 maximum low radiance value   PCDR B2 maximum high radiance value

PCDR B2 minimum low radiance value   PCDR B2 minimum high radiance value
PCDR B2 bandwidth
PCDR B3 maximum low radiance value   PCDR B3 maximum high radiance value
PCDR B3 minimum low radiance value   PCDR B3 minimum high radiance value
PCDR B3 bandwidth
PCDR B4 maximum low radiance value   PCDR B4 maximum high radiance value
PCDR B4 minimum low radiance value   PCDR B4 minimum high radiance value
PCDR B4 bandwidth
PCDR B5 maximum low radiance value   PCDR B5 maximum high radiance value
PCDR B5 minimum low radiance value   PCDR B5 minimum high radiance value
PCDR B5 bandwidth
PCDR B6 maximum low radiance value   PCDR B6 maximum high radiance value
PCDR B6 minimum low radiance value   PCDR B6 minimum high radiance value
PCDR B6 bandwidth
CTV B3
CTV (first) B5 and B6
CTV (second) B5 and B6
CTV B6 reflectance
CTV experimental data for B4 and B5 ratio limit
CTV experimental data for B4 and B2 ratio limit
CTV experimental data for B4 and B3 ratio limit
NDSI upper limit
NDSI lower limit
TEP conservative cloud percentage threshold
TEP desert percentage threshold
TEP snow percentage threshold
TEP high thermal effect threshold
TEP low thermal effect threshold
TEP maximum temperature separation from cloud

5.6 L0R Error Reporting Management

When the raw wideband data being processed contains many errors, the LPS L0R processing software can generate a large number of messages reporting that these errors have been encountered. The large number of messages can make it difficult to monitor LPS operations. Users can reduce the number of errors the LPS reports during L0R processing by setting any of a large number of error reporting thresholds.

When an error's threshold is set to a number greater than zero, the LPS L0R processing software will report errors only when the total number of errors in the processing run exceeds an integer multiple of the threshold. For example, if the threshold for CADU frame synchronization errors is set to 100, the LPS L0R processing software will report CADU frame synchronization errors when the total number of such errors exceeds 100, again when it exceeds 200, and so on.

This section describes how to view and edit the thresholds for LPS L0R error reporting.

5.6.1 Viewing or Editing L0R Error Reporting Thresholds

L0R thresholds may be viewed on individual strings. Unlike L0R parameters, changes must be made on each individual string and cannot be propagated from one string to the others. Table 5–4 describes the thresholds, and Table 5–5 lists suggested values and ranges.

To view or edit L0R thresholds:

1. Select the **Setup>View/Edit L0R Thresholds** command from the LPS GUI. The string currently in use will appear on the display along with the parameters.

2. Select the threshold in question.

3. To edit thresholds, insert the desired value in the field.

   If values are out of the permitted range in the database, you will not be allowed to move to a different field. A message at the bottom of the screen will notify you of the valid range of the field in question.

4. Click the OK button to keep changes or click the CANCEL button to leave parameters unchanged.

Table 5–4. Table and Attribute Names for L0R Error Thresholds (1 of 2)

Threshold	Table Name	Attribute Name
Raw Data Processing		
BCH Errors Modulo number of frames with uncorrectable errors in either the mission data or data pointer zones detected by BCH decoding allowed before operator notification	valid_rdp_thres	bch_thres
CADU Frame Synchronization Errors Modulo number of frames with synchronization pattern errors and flywheel frames detected allowed before operator notification	valid_rdp_thres	sync_thres
Cyclic Redundancy Code (CRC) Errors Modulo number of frames with errors detected by CRC decoding allowed before operator notification	valid_rdp_thres	crc_thres
Reed-Solomon Errors Modulo number of frames with uncorrectable errors detected by Reed-Solomon decoding allowed before operator notification	valid_rdp_thres	rs_thres

Table 5–4. Table and Attribute Names for L0R Error Thresholds (2 of 2)

Threshold	Table Name	Attribute Name
Major Frame Processing		
CADU Sequence Errors Number of ETM+ sequence counter errors allowed before operator notification for each error	valid_mfp_thres	mjf_cadu_seq_err_ thres
EOL Code Errors Number of ETM+ EOL errors allowed before operator notification for each error	valid_mfp_thres	eol_thres
Fully Filled Major Frames Number of entirely filled ETM+ major frames allowed before operator notification for each fully filled major frame	valid_mfp_thres	full_mjf_thres
Major Frame Synchronization Errors Number of ETM+ major frame synchronization errors allowed before operator notification for each error	valid_mfp_thres	mjf_sync_thres
Minor Frame Counter Errors Number of ETM+ minor frame counter errors allowed before operator notification for each error	valid_mfp_thres	mnf_ctr_thres
Timecode Errors Number of ETM+ timecode errors allowed before operator notification for each error	valid_mfp_thres	tc_thres
Partially Filled Major Frames Number of partially filled ETM+ major frames allowed before operator notification for each partially filled major frame	valid_mfp_thres	part_mjf_thres
PCD Processing		
Majority Vote Failures Modulo number of failed PCD majority votes allowed before operator notification	valid_pcd_thres	num_failed_votes
Missing Data Words Modulo number of missing PCD words allowed before operator notification	valid_pcd_thres	num_missing_ data_words

Table 5–5. Error Reporting Thresholds Suggested Values and Valid Ranges (1 of 2)

Threshold	Suggested Value	Valid Range	Description
Raw Data Processing			
CADU Frame Synchronization	1	0...2147483648	Modulo number of frames with synchronization pattern errors and flywheel frames detected allowed before operator notification
CRC	100	0...2147483648	Modulo number of frames with errors detected by CRC decoding allowed before operator notification
BCH	100	0...2147483648	Modulo number of frames with uncorrectable errors in either mission data or data pointer zones detected by BCH decoding allowed before operator notification
Reed-Solomon	100	0...2147483648	Modulo number of frames with uncorrectable errors detected by Reed-Solomon decoding allowed before operator notification
Major Frame Processing			
CADU Sequence Errors	20	0–99999999	Number of ETM+ sequence counter errors allowed before operator notification for each error
Minor Frame Counter Errors	20	0–99999999	Number of ETM+ minor frame counter errors allowed before operator notification for each error
Partially Filled Major Frames	100	0–99999999	Number of partially filled ETM+ major frames allowed before operator notification for each partially filled major frame
Major Frame Synchronization Errors	20	0–99999999	Number of ETM+ major frame synchronization errors allowed before operator notification for each error
EOL Code Errors	20	0–99999999	Number of ETM+ EOL errors allowed before operator notification for each error
Timecode Errors	20	0–99999999	Number of ETM+ timecode errors allowed before operator notification for each error
Fully Filled Major Frames	2	0–99999999	Number of entirely filled ETM+ major frames allowed before operator notification for each fully filled major frame

Table 5–5. Error Reporting Thresholds Suggested Values and Valid Ranges (2 of 2)

Threshold	Suggested Value	Valid Range	Description
PCD Processing			
Majority Vote Failures	1	1–99999999	Modulo number of failed PCD majority votes allowed before operator notification
Missing Data Words	1	1–99999999	Modulo number of missing PCD words allowed before operator notification

6. File Transfers to EDC DAAC

6.1 Introduction

This chapter instructs users on how to control and monitor EDC DAAC file transfers. It describes how to perform operations related to LPS L0R processing and how to

- View and edit LPS/EDC DAAC communication parameters stored in a string's database.

- Control automatic file availability notification.

- Determine the transfer status of a set of output files.

- Manage automatic output file deletion, including how to delete LPS output files manually.

6.2 LPS/EDC DAAC Communication Parameters Management

LPS/EDC DAAC communication parameters specify the host names, port numbers, user names, passwords, and other characteristics to be used in communications between the LPS and EDC DAAC.

NOTE

Each LPS string has its own set of LPS/EDC DAAC communication parameters. Parameter values must be set separately for each string.

6.2.1 Viewing or Editing the LPS/EDC DAAC Communication Parameters

LPS/EDC DAAC communications parameters (Table 6–1) may be edited and viewed through the GUI or SQL*Plus. Refer to Appendix E for SQL*Plus procedures. There are two groups of communication parameters. Most parameters may be viewed along with the rest of the L0R parameters. Refer to Section 5.5 for procedures.

The ECS Hardware String ID, ECS User ID, ECS Password, LPS Port Number, and ECS Port Number are in the second group. To see these parameters:

1. Select the **Setup>View/Edit Output File Transfer Config** command from the GUI.

2. Select the field in question.

3. To edit fields, insert the desired value in the field.

   If values are out of the permitted range in the database, you will not be allowed to move to a different field. A message at the bottom of the screen will notify you of the valid range of the field in question.

Table 6–1. Table and Attribute Names for LPS/EDC DAAC Communication Parameters (1 of 3)

Parameter	Attribute Name	Range/Suggested Value
LPS Port Number Port number on which DDN server listens for connection requests from EDC DAAC	lps_configuration.lps_ port_num	0..65535 NOTE: Port number should not be in use for any other Internet service on the LPS string.
LPS User ID User name to be used for authentication when sending messages to EDC DAAC host (not on GUI)	lps_configuration.lps_ user_id	20-character maximum length
LPS Password Password to be used for authentication when sending messages to EDC DAAC host (not on GUI)	lps_configuration.lps_ password	20-character maximum length
Earth Observing System (EOS) Core System (ECS) Hardware String ID Name of EDC DAAC host to which DANs should be sent in standard Internet name format (e.g., ecdaac2.cr.usgs.gov)	lps_configuration.ecs_ hardware_string_id	20-character maximum length
ECS Port Number Port number on which EDC DAAC host listens for connection requests from LPS	lps_configuration.ecs_ port_num	0..65535 NOTE: Port number should not be in use for any other Internet service on the EDC DAAC string.
ECS User ID User name to be used for authentication when sending messages to the EDC DAAC host	lps_configuration.ecs_ user_id	20-character maximum length
ECS Password Password to be used for authentication when sending messages to EDC DAAC host	lps_configuration.ecs_ password	20-character maximum length

Table 6–1. Table and Attribute Names for LPS/EDC DAAC Communication Parameters (2 of 3)

Parameter	Attribute Name	Range
Maximum Logon Attempts Number of authentication failures to allow for EDC DAAC connection attempts before failing	valid_ldt_parms.num_ auth_request	0..9999 suggested value: 1
Receive DAA Timeout Specifies amount of time to wait for receipt of a DAA before timing out (seconds)	valid_ldt_parms. timeout_receive_daa	0..9999 suggested value: 180 seconds
Send DAN Timeout Specifies amount of time to wait while attempting to send a DAN before timing out (seconds)	valid_ldt_parms. timeout_send_dan_ attempt	0..9999 suggested value: 4 seconds
Receive DDN Timeout Specifies amount of time to wait for receipt of a DDN before timing out (seconds)	valid_ldt_parms. timeout_receive_ddn	0..9999 suggested value: 30 seconds
Maximum Send DDA Attempts Number of times to attempt to send a DDA	valid_ldt_parms.num_ send_dda_attempt	0..9999 suggested value: 3
Send DAA Timeout Specifies amount of time to wait while attempting to send a DAA before timing out (seconds)	valid_ldt_parms. timeout_send_dda	0..9999 suggested value: 4 seconds
Socket Polling Interval Sleep time between socket reads (seconds)	valid_ldt_parms.read_ sleep_second	0..9999 suggested value: 1 second
Maximum Send DAN Attempts Number of times to attempt to send a DAN	valid_ldt_parms.read_ num_send_dan_ attempt	0..9999 suggested value: 3

Table 6–1. Table and Attribute Names for LPS/EDC DAAC Communication Parameters (3 of 3)

Parameter	Attribute Name	Range
LPS-ECS Save Message Flag Flag to save messages (not on GUI)	valid_ldt_parms.ldt_ecs_msg_save	0 = no, 1 = yes
Authentication Request Timeout Specifies amount of time to wait for acknowledgment of an authentication request before timing out (seconds)	valid_ldt_parms.read_timeout_auth_request	0..9999 suggested value: 2 seconds

4. Click the OK button to keep changes or click the CANCEL button to leave parameters unchanged.

The parameters LPS User ID and LPS Password cannot be viewed or edited through the LPS GUI. The LPS Hardware String is viewed on every panel, but cannot be edited through the GUI.

6.3 Automatic File Availability Notification Management

Normally, the LPS will automatically notify EDC DAAC that output files are available when L0R processing completes successfully. Users can, however, disable this feature and re-enable it later. While automatic file availability notification is disabled, a DAN is not sent to EDC DAAC whenever L0R processing produces a set of output files.

NOTE

The state of automatic notification is preserved even when the LPS system is shut down. When you bring the LPS software up again, the state (either enabled or disabled) will be the same as when it was last brought down.

When automatic notification is enabled after having been disabled, the LPS does not notify EDC DAAC of any file sets produced while automatic notification was disabled.

The state of automatic notification is independent on each LPS string. Enabling or disabling automatic notification on one string does not affect its state on other strings.

To disable automatic notification to EDC DAAC of output file availability:

1.   Select the **FileMgmt>DAN Transfer State...** command from the LPS GUI. The "Set DAN Transfer State" dialog appears.

2.   Click the disable option.

3.   Click the OK button.

To enable automatic notification to EDC DAAC of output file availability:

1.   Select the **FileMgmt>DAN Transfer State...** command from the LPS GUI. The "Set DAN Transfer State" dialog appears.

2.   Click the enable option.

3.   Click the OK button.

To monitor automatic notification, view the status and error messages generated by the senddan process, which notifies EDC DAAC that output files are available for transfer. To view status and error messages generated by senddan, set up a real-time display of the LPS Journal file (see Section 2.7.2).

6.4 DDN Server Management

The DDN server, rcvddn, accepts and processes DDNs from EDC DAAC. Users can control automatic DDN receipt and processing by terminating or restarting the DDN server.

NOTE

There is a separate DDN server on each LPS string. Changing the status of the DDN server on one string does not affect the status of the DDN servers on the other strings.

6.4.1 Checking Whether the DDN Server Is Active

To determine whether the DDN server is active, select **FileMgmt>Start DDN Server**. If the DDN server is active, a dialog will appear telling the operator that the DDN server is already started.

6.4.2 Stopping the DDN Server

If a DDN is currently under processing by the DDN server, the DDN server will not be stopped until the DDN server completed DDN processing. To stop the DDN server, select the **FileMgmt>Stop DDN Server...** command and click the OK button in the confirmation dialog. It will not take effect until current DDN is done processing.

6.4.3 Starting the DDN Server

To start the DDN server, select the **FileMgmt>Start DDN Server...** command and click the OK button in the confirmation dialog.

6.5 Checking the Status of a Set of Output Files

To determine the status of a set of output files, first determine whether the LPS has notified EDC DAAC that the files are available (by sending a DAN) and whether EDC DAAC has acknowledged receipt of the notification (by returning a DAA). If EDC DAAC has been notified and has acknowledged, determine whether or not the files have been transferred successfully.

6.5.1 Checking Whether EDC DAAC Has Been Notified

To determine whether EDC DAAC has been notified that a set of output files is ready for transfer:

1. Find the LPS database contact ID and the file version number for the processing instance of interest. To learn how, read Appendix E, Sections E.1.1 and E.1.2.

2. Determine whether a DAN was sent for the set of output files by typing the following to SQL*Plus:

    ```
    SQL> SELECT dan_status FROM ldt_dan_info
    2>    WHERE contact_sequence_id = Contact Identifier
    3>    AND file_version_number = File Version Number ;
    ```

    Contact Identifier is the contact ID for the contact. *File Version Number* is the file version number.

3. Use Table 6–2 to look up the meaning of the value reported by SQL*Plus.

Example: Determine the status of the first instance of processing the contact scheduled to start on day 211 of 1998 at 09:07:36.

1. Using one of the methods described in Section E.1.2, the contact ID is determined to be 86214. The file version number is 0 because this is the first instance of processing the contact.

2. Determine whether a DAN was sent for the set of output files by typing the following to SQL*Plus:

    ```
    SQL> SELECT dan_status FROM ldt_dan_info
    2>    WHERE contact_sequence_id = 86214
    3>    AND file_version_number = 0;
    ```

    Use Table 6–2 to look up the meaning of the value reported by SQL*Plus.

Table 6–2. Meaning of dan_status Attribute Values

Value	Meaning
0	Prepare DDN to be sent
1	EDC DAAC has not been notified because automatic notification was disabled when this set of output files was ready for transfer.
2	EDC DAAC has been notified, and the LPS is waiting for acknowledgment from EDC DAAC.
3	EDC DAAC has not been notified because DDN cannot be sent.
4	EDC DAAC has been notified and has also acknowledged the notification.
5	EDC DAAC has been notified, has not acknowledged, and the maximum waiting time for an acknowledgment has been exceeded.
6	The DAN was canceled, and the EDC DAAC will not be notified.

6.5.2 Checking Whether EDC DAAC Has Transferred the Files

A basic status of the output files may be checked through the output file management as described in Section 6.8.1. More details are available through the IRIX as described in this section.

To determine whether EDC DAAC has transferred a set of files:

1. Find the LPS database contact ID and the file version number for the processing instance of interest. To learn how to do so, read Sections E.1.1 and E.1.2.

2. Verify that EDC DAAC has been notified of its availability as described in Section 6.4.1. If EDC DAAC has not been notified that the files are available, EDC DAAC has not yet transferred the files.

3. Retrieve the status of the output files from the LPS database by typing the following to SQL*Plus:

   ```
   SQL> SELECT file_set_transfer_status FROM ldt_file_set_info
   2>    WHERE contact_sequence_id = Contact Identifier
   3>    AND file_version_number = File Version Number;
   ```

   Contact Identifier and *File Version Number* are the contact ID and file version number, respectively, determined in step 1.

4. Use Table 6–3 to look up the meaning of the value reported by SQL*Plus.

Table 6–3. Meaning of file_set_transfer_status Attribute Values

Value	Meaning
0	The files have been successfully transferred.
1	EDC DAAC failed in its last attempt to transfer any files. No files have been transferred.
2	EDC DAAC has not yet sent a DDN. No files have been transferred.
3	EDC DAAC has successfully transferred some of the files. Some of the files have not been transferred.
4	Unknown file set transfer status.

5. If the value of file_set_transfer_status is 3 (some of the files have been successfully transferred and others have not), you can determine the disposition of each file by typing the following to SQL*Plus:

```
SQL> COLUMN file_path HEADING 'Path' FORMAT A25
SQL> COLUMN file_name HEADING 'Name' FORMAT A22
SQL> COLUMN file_xfer_disposition HEADING 'Disp' FORMAT 000
SQL> SELECT file_path, file_name, file_xfer_disposition
3>    FROM ldt_file_set_info f, sub_intv s, lps_file_info i
4>    WHERE f.contact_sequence_id = 86214
5>    AND f.file_version_number = 0
6>    AND f.contact_sequence_id = s.contact_sequence_id
7>    AND f.file_version_number = s. file_version_number
8>    AND s.sub_intv_sequence_id = i.sub_intv_sequence_id
9>    ORDER BY file_path, file_name;
```

6. Use Table 6–4 to look up the meaning of the file_transfer_disposition reported by SQL*Plus for each file.

Example: Determine whether the file set in the previous example has been transferred to EDC DAAC.

1. Using one of the methods described in Section E.1.2, the contact ID is determined to be 86214. The file version number is 0 because this is the first instance of processing the contact.

Table 6–4. Meaning of file_xfer_disposition Attribute Values

Value	Meaning
0	File has been successfully transferred
1	Network failure
2	Unable to establish FTP connection
4	All file groups/files not found
5	FTP failure – too many errors in file transfer
6	Post-transfer doublecheck failed
7	FTP command error
242	Duplicate file name in granule
243	Metadata processing error
244	Resource allocation failure
245	Ingest software internal error
246	Database access error
247	Correct number of metadata files
248	Correct number of science files
249	Correct number of files
250	Data conversion failure
251	Request canceled
252	Unknown data type
253	Invalid or missing data type
254	File input/output error
255	Data archive error

2. As determined in the previous example, EDC DAAC has been notified and has acknowledged.

3. Type the following to SQL*Plus:

```
SQL> SELECT file_set_transfer_status FROM ldt_file_set_info
```

```
2>    WHERE contact_sequence_id = 86214

3>    AND file_version_number = 0;

file_set_transfer_status

------------------------

                        3
```

4. In Table 6–3, a file_set_transfer_status value of 3 indicates that some of the files have been transferred and others have not.

5. Type the following to SQL*Plus:

```
SQL> COLUMN file_path HEADING 'Path' FORMAT A25

SQL> COLUMN file_name HEADING 'Name' FORMAT A22

SQL> COLUMN file_xfer_disposition HEADING 'Disp' FORMAT 000

SQL> SELECT file_path, file_name, file_xfer_disposition

3>    FROM ldt_file_set_info f, sub_intv s, lps_file_info i

4>    WHERE f.contact_sequence_id = 86214

5>    AND f.file_version_number = 0

6>    AND f.contact_sequence_id = s.contact_sequence_id

7>    AND f.file_version_number = s. file_version_number

8>    AND s.sub_intv_sequence_id = i.sub_intv_sequence_id

9>    ORDER BY file_path, file_name;

Path                                              Name                    Disp

------------------------------------------------  ----------------------  ----

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.B60    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.B70    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.B81    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.B82    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.CAL  254

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.MSD    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.MTA    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109000.PCD    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109010.B60    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109010.B70    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109010.B81    5

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109010.B82    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109010.CAL    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109010.MSD    0

/u01/st/outfile/contact-86214/ver-0/subint-3624  L71EDC219821109010.MTA    0
```

```
        /u01/st/outfile/contact-86214/ver-0/subint-3624      L71EDC219821109010.PCD      0
```

6. Table 6–4 indicates that all files have transferred except for L71EDC219821109000.CAL, which failed because of file input/output errors at EDC DAAC, and L71EDC219821109010.B81, which failed because of FTP errors.

6.5.3 Resending DANs

A DAN that has had problems arriving at the EDC DAAC may be resent. To resend a suspended DAN:

1. Select the **FileMgmt>Resend Suspended DAN...** command from the LPS GUI. The "Resend Suspended DAN" dialog appears.

2. Click the OK button to resend the DAN

To resend a failed DAN:

1. Select the **FileMgmt>Resend Failed DAN...** command from the LPS GUI. The "Resend Failed DAN" dialog appears. DAN sequence numbers are displayed, along with their contact start and stop times.

2. Select the desired DAN.

3. Click the OK button to resend the DAN.

6.6 Checking Whether EDC DAAC Is Transferring Files

To determine whether EDC DAAC is currently transferring files from a string:

1. Find the EDC DAAC host name in the LPS database by typing the following to SQL*Plus:

   ```
   SQL> SELECT ecs_hardware_string_ID FROM lps_configuration;
   ```

2. Determine whether there is an active FTP connection by typing the following to an IRIX shell:

   ```
   % netstat | grep EDC DAAC Host Name | grep ftp
   ```

3. If the FTP connection to the EDC DAAC host is active, this command will produce a line describing the state of the connection:

   ```
   tcp 0 0 lps001.cr.edc.gov.ftp edcdaac1.cr.edc.gov.1098 ESTABLISHED
   ```

4. For a description of the line's meaning, consult the netstat(1) section of *IRIX 5.3 Reference Manual* (Reference 2).

6.7 Checking the Contents of LPS/EDC DAAC Messages

Normally, LPS/EDC DAAC communication messages are processed and discarded. However, the LPS software is able to save all LPS/EDC DAAC communications messages for review.

NOTE

> Saving LPS/EDC DAAC communication messages is controlled independently on each LPS string. You must set the LDT_ECS_MSG_SAVE flag in valid_LDT_parms table environment variable on each string on which you want to save messages.

To activate automatic saving of LPS/EDC DAAC communication messages:

1. Type the following to SQL*Plus:

   ```
   SQL>UPDATE valid_ldt_parms SET ldt_ecs_msg_save = 1;
   ```

2. Verify that the update occurred correctly by noting that the following message appears:

   ```
   1 row updated
   ```

3. Save the change by typing the following to SQL*Plus:

   ```
   SQL>COMMIT;
   ```

4. Verify that the changes have been saved by noting that the following message appears:

   ```
   Commit complete.
   ```

To disable automatic saving of LPS/EDC DAAC communication messages:

1. Type the following to SQL*Plus:

   ```
   SQL>UPDATE valid_ldt_parms SET ldt_ecs_msg_save = 0;
   ```

2. Verify that the update occurred correctly by noting that the following message appears:

   ```
   1 row updated
   ```

3. Save the change by typing the following to SQL*Plus:

   ```
   SQL>COMMIT;
   ```

4. Verify that the changes have been saved by noting that the following message appears:

   ```
   Commit complete.
   ```

When automatic LPS/EDC DAAC communications saving is enabled, the LPS software will write DANs, DAAs, and authentication messages to the directory indicated by the LPS_DANFILE_PATH environment variable (normally "$LPS_HOME/DAN"). It will write DDNs, DDAs, and authentication messages to the directory indicated by the LPS_DDNFILE_PATH environment variable (normally "$LPS_HOME/DDN"). The file names have the format

TTT_NNNNNNNNNN_MM.msg

where TTT is the message type (DAN, DAA, DDN, or DDA), NNNNNNNNNN is the DAN ID associated with the communication, and MM is an incrementing message number. To determine the DAN ID associated with a particular instance of L0R output files:

1. Find the contact ID and the file version number of the L0R processing instance of interest. To learn how to do so, read Appendix E, Sections E.1.1 and E.1.2.

2. Type the following to SQL*Plus:

```
SQL> SELECT dan_sequence_num FROM ldt_dan_info
2>   WHERE contact_sequence_id = Contact Identifier
3>   AND file_version_number = File Version Number;
```

LPS/EDC DAAC messages are saved in American Standard Code for Information Interchange (ASCII) files. The files can be viewed using any of the IRIX file display utilities, such as more(1) or page(1) and may be modified using any text editor, such as vi. See Reference 2 for details on these utilities.

6.8 Managing Output File Deletion

This section describes how to control the LPS software's automatic file deletion capabilities. It describes how to mark file sets for retention so that they will not be deleted automatically. It also describes how to delete LPS output files manually.

NOTE

> The LPS software may delete a subset of the files from an instance of L0R processing, but only entire file sets can be marked for retention.

Root permission is necessary to delete files.

6.8.1 Output File Retention and Deletion

Output files can be marked for retention and deletion or deleted immediately through the LPS GUI. The **Start L0R Processing** GUI has a selection to delete output files after transfer. Changes to this selection may be made through **Output File Set Management**.

To mark output files for retention and deletion:

1. Select **FileMgmt>Output File Set Management** from the LPS GUI.

   The display will list the capture string ID, the contact start and stop times, the current file version number, and current transfer, retention, and deletion decision status. The transfer status has three values: yes – all files from the contact transferred, partial – some files transferred, and no – no files transferred.

2. Select the contact file version number combination desired.

3. To retain the output files after transfer, select the retain after transfer box. The retain after transfer will have an X in its column for retention. Otherwise, the files will be

deleted after transfer. Unselect the box to change this decision or click the CANCEL button before hitting the OK button.

4. To delete the outfiles immediately, select the delete output file set box. A query form will appear asking if you really want to delete the files. Click the OK button on that form if desired. If no files have been transferred, an A will appear in the delete now column. If only some of the files have been transferred (a partial transfer status), another form appears, asking to delete only the transferred files or all the files. If all is selected, an A will appear in the column. If only the transferred files, a T will appear in the column.

5. Click the OK button to act on the choices made.

6.9 Deleting L0R Output Files Manually

The LPS software maintains a record of the deletion status (deleted or not deleted) for every output file in the LPS database. When you delete output files, you must also make the appropriate updates to the LPS database.

```
┌─────────────────┐
│    WARNING      │
└─────────────────┘
```

Deleting L0R output files without making the appropriate updates in the LPS database will render the LPS database inconsistent and may cause run-time errors in LPS software.

You can delete an output file set, an output file group, or individual output files.

6.9.1 Deleting a File Set

To delete a set of output files:

1. Find the LPS database contact ID and the file version number for the processing instance of interest. To learn how to do so, read Appendix E, Sections E.1.1 and E.1.2.

2. Delete all files in the set by typing the following to an IRIX shell:

```
% cd $LPS_OUTFILE_PATH/Contact Identifier/File Version Number
% rm -r *
```

3. Mark each file group in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_group_info
2>    SET file_group_deletion_status = 2
3>    WHERE sub_intv_sequence_id IN
4>    (SELECT sub_intv_sequence_id FROM sub_intv
5>      WHERE contact_sequence_id = Contact Identifier
6>      AND file_version_number = File Version Number);
```

4. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

5. Mark each file in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_info
2>    SET file_deletion_status = 1
3>    WHERE sub_intv_sequence_id IN
4>    (SELECT sub_intv_sequence_id FROM sub_intv
5>      WHERE contact_sequence_id = Contact Identifier
6>      AND file_version_number = File Version Number);
```

6. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

7. Commit the updates by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

8. Verify that the commit completed successfully by noting that the following message appears:

```
Commit complete.
```

Example: Delete the file set produced by the first instance of processing the contact scheduled to start on day 211 of 1998 at 09:07:36.

1. Using one of the methods described in Section E.1.2, the contact ID is determined to be 86214. The file version number is 0 because this is the first instance of processing the contact.

2. Delete all files in the set by typing the following to an IRIX shell:

```
% cd $LPS_OUTFILE_PATH/86214/0
% rm -r *
```

3. Mark each file group in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_group_info
2>    SET file_group_deletion_status = 1
3>    WHERE sub_intv_sequence_id IN
4>    (SELECT sub_intv_sequence_id FROM sub_intv
5>      WHERE contact_sequence_id = 86214
6>      AND file_version_number = 0);
1 row updated.
```

4. The message "1 row updated." indicates that the update occurred correctly.

5. Mark each file in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE lps_file_info
2>    SET file_deletion_status = 2
3>    WHERE sub_intv_sequence_id IN
4>    (SELECT sub_intv_sequence_id FROM sub_intv
5>      WHERE contact_sequence_id = 86214
6>      AND file_version_number = 0);
1 row updated.
```

6. The message "1 row updated." indicates that the update occurred correctly.

7. Commit the updates by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

8. The message "Commit complete." indicates that the update has been saved to the database.

6.9.2 Deleting a File Group

> **WARNING**
>
> Deleting the last file group of a file set without updating the file set information in the database can leave the LPS database in an inconsistent state. If you are deleting the last file group of a file set, follow the procedure for deleting a file set rather than the procedure defined here.

To delete a file group:

1. Find the LPS database contact ID and the file version number for the processing instance of interest. To learn how to do so, read Appendix E, Sections E.1.1 and E.1.2.

2. Find the LPS database subinterval ID for the file group of interest. To learn how to retrieve the subinterval ID from the database, read Section E.1.3.

3. Delete all files in the set by typing the following to an IRIX shell:

```
% cd $LPS_OUTFILE_PATH/Contact Identifier/File Version Number \
? /Subinterval Identifier
% rm -r *
```

4. Mark each file group in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_group_info
2>    SET file_group_deletion_status = 1
3>    WHERE sub_intv_sequence_id = Subinterval Identifier;
```

5. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

6. Mark each file in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_info
2>    SET file_deletion_status = 1
3>    WHERE sub_intv_sequence_id = Subinterval Identifier;
```

7. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

8. Commit the updates by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

9. Verify that the commit completed successfully by noting that the following message appears:

```
Commit complete.
```

Example: Delete the file group for the first subinterval in the first instance of processing the contact scheduled to start on day 211 of 1998 at 09:07:36.

1. Using one of the methods described in Section E.1.1, the contact ID is determined to be 86214. The file version number is 0 because this is the first instance of processing the contact.

2. Using one of the methods described in Section E.1.3, the subinterval ID is determined to be 341576.

3. Delete all files in the set by typing the following to an IRIX shell:

```
% cd $LPS_OUTFILE_PATH/86214/0/341576
% rm -r *
```

4. Mark each file group in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_group_info
2>    SET file_group_deletion_status = 2
3>    WHERE sub_intv_sequence_id = 341576;
1 row updated.
```

5. The message "1 row updated." indicates the update occurred correctly.

6. Mark each file in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_info
2>    SET file_deletion_status = 1
3>    WHERE sub_intv_sequence_id = 341576;
1 row updated.
```

7. The message "1 row updated." indicates the update occurred correctly.

8. Commit the updates by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

9. The message "Commit complete." indicates the commit occurred correctly.

6.9.3 Deleting Individual Files

```
┌─────────────────────┐
│      WARNING        │
└─────────────────────┘
```

Deleting the last file of a file group without updating the file group information in the database can leave the LPS database in an inconsistent state. If you are deleting the last file of a file group, follow the procedure for deleting a file group rather than the procedure defined here.

To delete an LPS output file:

1. Find the LPS database contact ID and the file version number for the processing instance of interest. To learn how to do so, read Appendix E, Sections E.1.1 and E.1.2.

2. Find the LPS database subinterval ID for the file group of interest. To learn how to retrieve the subinterval ID from the database, read Section E.1.3.

3. Delete the file by typing the following to an IRIX shell:

```
% rm $LPS_OUTFILE_PATH/Contact Identifier/File Version Number \
? /Subinterval Identifier/File Name
```

4. Mark the file as deleted in the LPS database by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_info
2>    SET file_deletion_status = 2
3>    WHERE sub_intv_sequence_id = Subinterval Identifier
4>    AND file_name = 'File Name';
```

5. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

6. Commit the update by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

7. Verify that the commit completed successfully by noting that the following message appears:

```
Commit complete.
```

Example: Delete the file L71EDC219821109010.CAL for the first subinterval in the first instance of processing the contact scheduled to start on day 211 of 1998 at 09:07:36.

1. Using one of the methods described in Section E.1.1, the contact ID is determined to be 86214. The file version number is 0 because this is the first instance of processing the contact.

2. Using one of the methods described in Section E.1.3, the subinterval ID is determined to be 341576.

3. Delete the file in the set by typing the following to an IRIX shell:

```
% rm $LPS_OUTFILE_PATH/86214/0/341576/L71EDC219821109010.CAL
```

4. Mark the file group in the set as deleted by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_info
2>    SET file_deletion_status = 1
3>    WHERE sub_intv_sequence_id = 86214
4>    AND file_name = 'L71EDC219821109010.CAL';
1 row updated.
```

5. The message "1 row updated." indicates the update occurred correctly.

6. Commit the updates by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

7. The message "Commit complete." indicates the commit occurred correctly.

7. Report Generation

This section discusses the three reports that may be generated through the LPS GUI.

7.1 Data Receive Summary Report

The Data Receive Summary Report supplies information about the data received in a contact. Figure 7–1 illustrates a sample report. The report includes the capture string, the capture source, the filename to which the raw data was saved; the scheduled and actual contact start and stop times; the scheduled, expected, and received volume of data in megabytes and number of scenes; and the transmission rate in megabytes per second.

To generate the report:

1. Select the **Report>Data Receive Summary** command from the LPS GUI.

2. From the list of start and stop times, select the contact period in question.

3. Click the OK button to generate the report.

7.2 LPS Quality and Accounting Report

The LPS Quality and Accounting Report supplies Q&A information reported by subsystems on a subinterval level for a contact and compares to what the LPS calculates for the contact as a whole. Figure 7–2 illustrates a sample report. The discard line is the difference between the total and the sum of the subintervals. The captured raw data file, capture string, contact start and stop times, and data volume in megabytes are supplied. Other information supplied for each subinterval and for the contact include

- Subinterval start and stop times

- Number of CADUs

- Number of megabytes

- Number of scenes

- Total major frames

- Number of partially filled major frames

- Number of fully filled major frames

- Quality information about CADU processing

- Bit error rate

- Number of imagery timecode errors
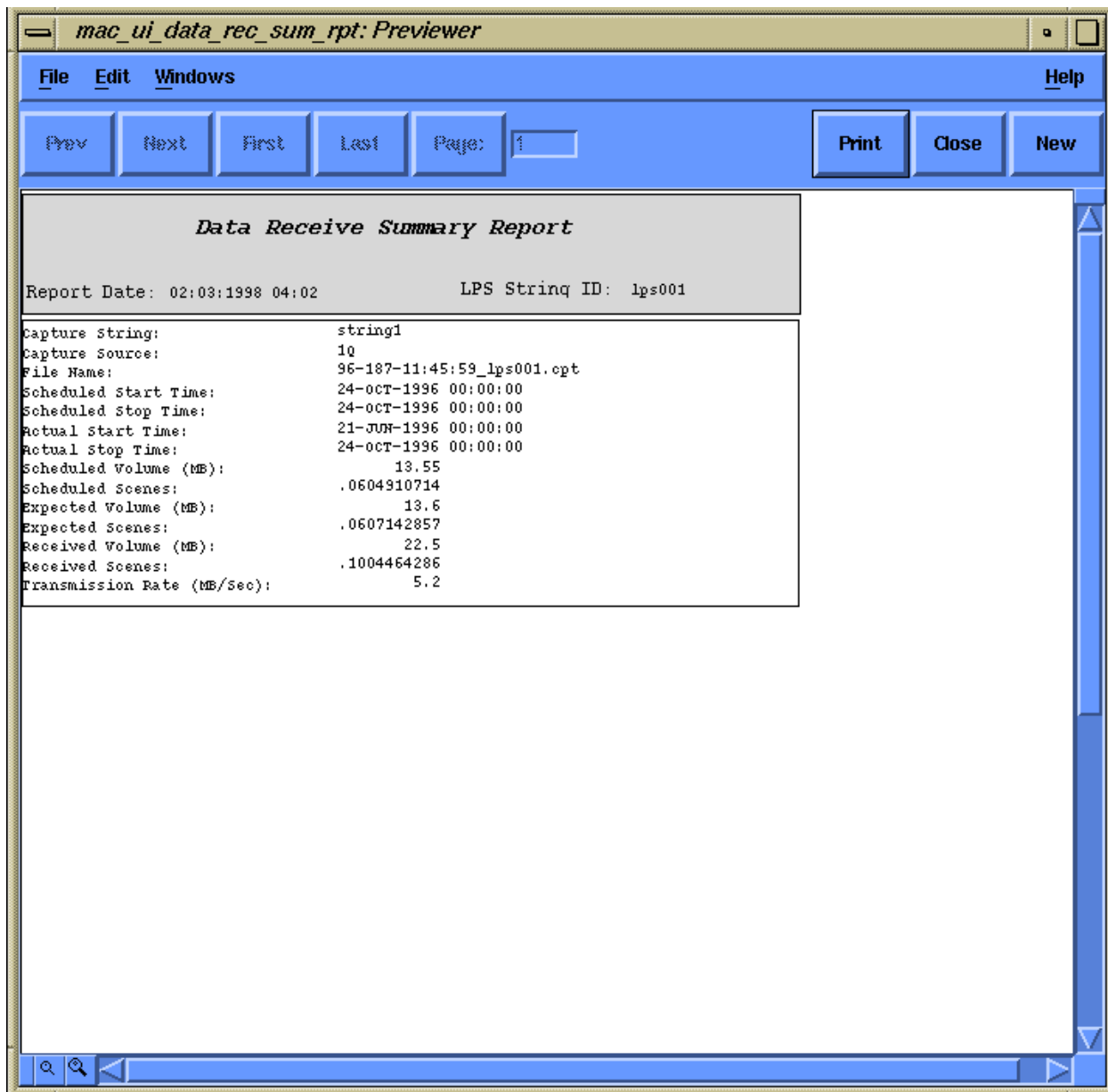
Figure 7–1. Data Receive Summary Report Sample

mac_ui_lps_qa_rpt: Previewer

File   Edit   Windows   Help

Prev   Next   First   Last   Page: 1   Print   Close   New

Report Date: 02:03:1998 04:02   *LPS Quality & Accounting Repo*

```
Capture File:          97-193-16:15:00_tc420.data
Capture String:        string1
Contact Start:         06:20:1997 00:00:00
Contact Stop:          06:22:1997 00:00:00
Data Volume (MBytes):  10.5
```

SI No	Subinterval Start	Subinterval Stop	CADUs	MBytes	Scenes	MJF Total	MJF Part Fill	MJF All Fill	Sync Errs
0	1997:172:07:22:53.3120000	1997:172:07:23:25.9226250	123455	127.899	4	0	20	34	0
1	1997:172:07:22:53.3120000	1997:172:07:23:25.9226250	123555	128.003	3	0	44	30	0
DISCARD			-192010	-198.92					90
TOTAL			55000	56.98					90

Figure 7–2. LPS Quality and Accounting Report Sample (1 of 2)

File   Edit   Windows   Help

| Prev | Next | First | Last | Page: | 1 | | Print | Close | New |

ort   LPS String ID: lps001

Tolerances

Search: 2
Check: 5
Bit Slip Extent: 1
Flywheel: 1
Check Error: 2
Lock Error: 4

CADU Quality

Flywhl	Missing	CRC ERR	RS Corr Fmt1	RS Corr Fmt2	RS Uncorr	BCH Corr	BCH Corr Pntr	BCH Uncorr	BCH Uncorr Pntr	BER	Time Code Err
1200	0	100	339	0	0	0	0	0	0	1.16	0
2300	0	200	0	500	0	0	0	0	0	.416	0
-1200		110				0	0	0	0		
2300		410			0	0	0	0			

Figure 7–2. LPS Quality and Accounting Report Sample (2 of 2)

The tolerances in CADU processing are supplied. CADU quality information includes the number of

- Synchronization errors

- CADUs flywheeled

- CADUs missing

- CRC errors

- Reed-Solomon corrected in format 1

- Reed-Solomon corrected in format 2

- Reed-Solomon uncorrected

- BCH corrected and uncorrected in data zone and pointer

To generate the report:

1. Select the **Report>LPS Quality/Accounting** command from the LPS GUI.

2. Select the contact period with file version of interest from the start and stop times and file version numbers.

3. Click the OK button to generate the report.

7.3 LPS Periodic Quality and Accounting Report

The LPS Periodic Quality and Accounting Report provides return-link Q&A information reported by subsystems on a segment basis for a selected contact period. Figure 7–3 illustrates a sample report. The captured raw data file, capture string, contact start and stop times, and the data volume in megabytes are supplied. The tolerances in CADU processing for the contact period are supplied. For each segment, the following information is supplied:

- Number of CADUs

- Number of megabytes

- CADU quality

- Bit error rate

CADU quality information includes the count of

- CADUs with synchronization errors

- Flywheel CADUs

- Missing CADUs

- Both correctable and uncorrectable VCDU headers (Reed-Solomon checked)

• CADUs with BCH errors both corrected and uncorrected for the mission data zone

```
┌─────────────────────────────────────────────────────────────────────────────────────────┐
│  ▭   mac_ui_lps_p_qa_rpt: Previewer                                              ▫  ☐    │
├─────────────────────────────────────────────────────────────────────────────────────────┤
│   File   Edit   Windows                                                            Help   │
├─────────────────────────────────────────────────────────────────────────────────────────┤
│  ┌──────┐ ┌──────┐ ┌──────┐ ┌──────┐ ┌──────┐ ┌──────┐        ┌──────┐ ┌──────┐ ┌──────┐ │
│  │ Prev │ │ Next │ │ First│ │ Last │ │ Page:│ │1     │        │ Print│ │ Close│ │ New  │ │
│  └──────┘ └──────┘ └──────┘ └──────┘ └──────┘ └──────┘        └──────┘ └──────┘ └──────┘ │
├─────────────────────────────────────────────────────────────────────────────────────────┤
```

LPS Periodic Quality & Accouting Report

Report Date: 02:03:1998 04:02

Capture File: 97-193-16:15:00_tc420.data
Capture String: string1
Contact Start: 06:20:1997 00:00:00
Contact Stop: 06:22:1997 00:00:00
Data Volume (MBytes): 10.5

CADU Quality

Segment ID	CADUs	MBytes	Sync Errs	Flywhl	Fill	CRC ERR	RS Corr Fmt1	RS Corr Fmt2	RS Uncorr	BCH Corr Data Zn
0	25000	25.9	33	900	99	240	0	0	0	0
1	30000	31.08	57	1400	57	170	0	0	0	0

Figure 7–3. LPS Periodic Quality and Accounting Report Sample (1 of 2)

LPS String ID: lps001

Tolerances

Search:	2
Check:	5
Bit Slip Extent:	1
Flywheel:	1
Check Error:	2
Lock Error:	4

BCH Corr Pntr	BCH Uncorr Data Zn	BCH Uncorr Pntr	BER
0	0	0	24.13
0	0	0	28.96
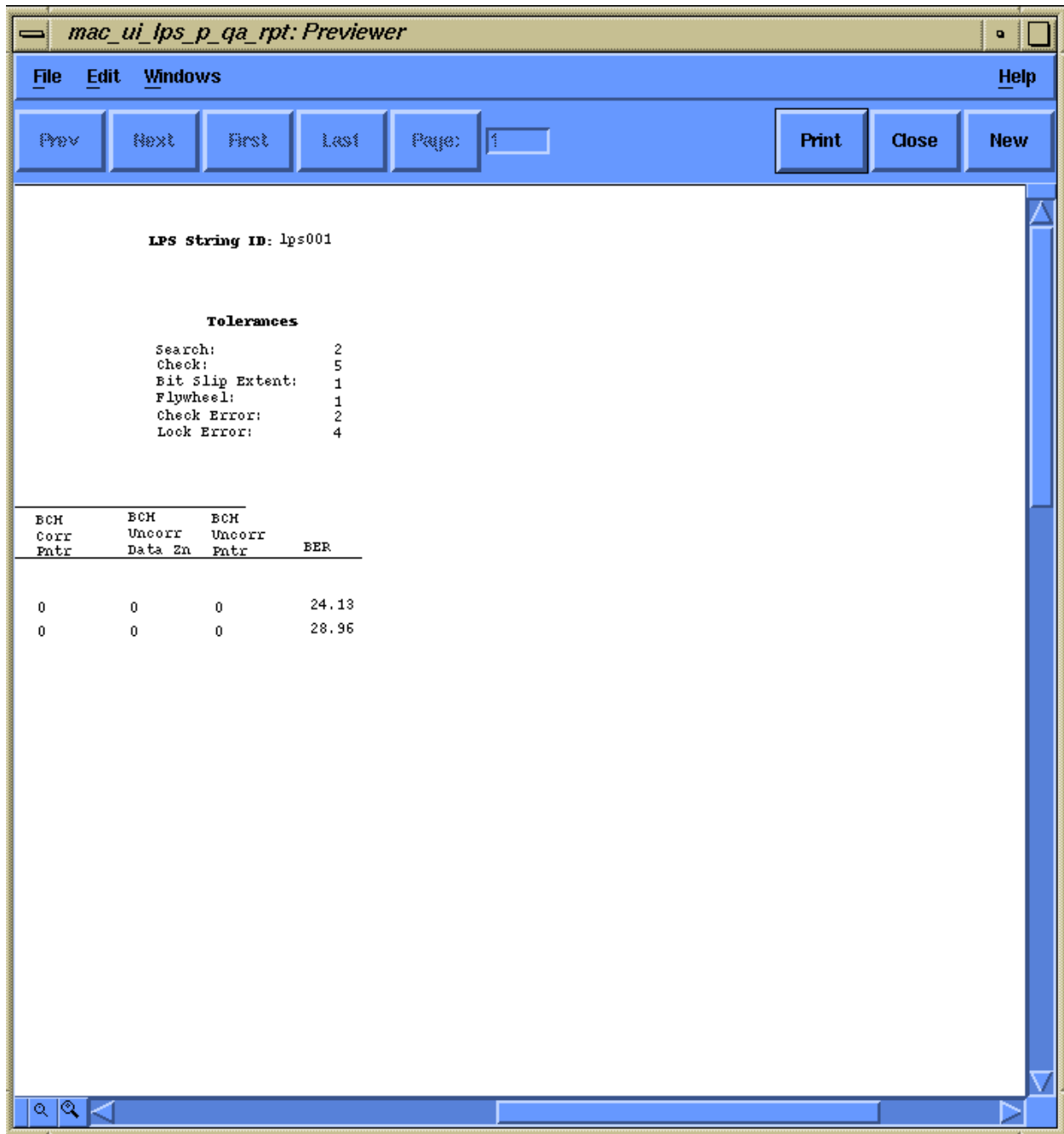
Figure 7–3. LPS Periodic Quality and Accounting Report Sample (2 of 2)

- CADUs with BCH errors both corrected and uncorrected for the data pointer zone

- CADUs with CRC errors

To generate the report

1. Select the **Report>Periodic Quality/Accounting** command from the LPS GUI.

2. Select the contact period with the file version number of interest from the selection list displayed.

3. Click the OK button to generate the report.

7.4 File Transfer Summary Report

The File Transfer Summary Report supplies information about which files were transferred from data that was captured over the report period. Figure 7–4 illustrates a sample report. Information supplied in the report includes the timespan of the report; the number of band, PCD, MSCD, browse, metadata, and calibration files that are available; the number of these files that were transmitted and retained; the number of files that were transmitted and deleted; and the total number of files available, transmitted and retained, and transmitted and deleted. It also supplies a list of available and transmitted filenames and the amount of available space.

To generate the report:

1. Select the **Report>File Transfer Summary** command from the LPS GUI.

2. Enter the begin and end times of interest in the format MM:DD:YYYY HH:MM:SS.

3. Click the OK button to generate the report of all files transferred (that were captured) during this time.

File   Edit   Windows   Help

Prev   Next   First   Last   Page: 1   Print   Close   New

LPS File Transfer Report

Report Date: 02:03:1998 16:11:23

Report Period:
       Begin 01:03:1997 00:00:00
        End 02:04:1998 00:00:00

	Available For Transmission	Transmitted And Deleted
Band	0	0
PCD	0	0
MSCD	0	0
Browse	0	0
Metadata	0	0
Calibration	0	0
Totals	0	0

Volume (MB)

Available for Transmission File Names   Transmitted and Deleted File N

Figure 7–4. File Transfer Summary Report Sample (1 of 2)

File   Edit   Windows   Help

Prev   Next   First   Last   Page: 1   Print   Close   New

sfer Report

LPS String Id: lps001

Available Space (MB): 1553

ted And Deleted   Transmitted And Retained

0

0

0

0

0

0

0

ted and Deleted File Names   Transmitted and Retained File Names

Figure 7–4. File Transfer Summary Report Sample (2 of 2)

8. Testing the LPS

The LPS will regularly need to be tested to verify the impact of the IAS CPF or changes in other L0R parameters. It also will be tested due to software changes and maintenance on hardware.

The method to test the system is to send through for processing a raw data file that has been previously processed. The transmit device is hardwired to a specific string to transmit the data. If it is not connected, the data will "fall on the floor." Root permission is necessary to run this function.

To send data:

1. Select the **Test>Send Data** command from the LPS GUI.

2. From the list of available data files, select the file of interest.

3. If you desire to dedicate a processor to sending this data, select the isolate processor key.

4. If you desire to suspend L0R processing during sending of this data, select the suspend L0R processing key.

5. Click the OK button to proceed.

To receive the test data, the receiving string should use the procedure to capture data that is described in Section 3.3.1.

One utility to assist in testing the LPS is a feature in the management and control subsystem (MACS) that allows a user to turn off the database rollback and file deletion, which occurs after unsuccessful runs of the LPS.

To turn off the rollback, the user needs to define the environment variable **NOROLLBACK** in the current shell. It does not need to be assigned to anything, just defined. If this environment variable is defined in the shell, MACS will not perform the rollback or file deletions and will write a warning message to the Journal to remind the user that this MACS feature has been turned off. If this environment variable is <u>not</u> defined, MACS will continue to rollback the database and delete all LPS output files after all runs that terminated with failure conditions.

9. LPS Maintenance

9.1 Introduction

This chapter explains how to perform routine LPS maintenance activities. These activities include backing up the LPS database; purging the LPS database of obsolete records; purging the LPS Journal file; and deleting unneeded trouble files, saved LPS/EDC DAAC messages, and saved reports. Maintenance functions should not be performed while any LPS processing is in progress.

This chapter does not explain how to perform hardware and system software maintenance or hardware or software upgrades. It also does not include detailed instructions for recovering from catastrophic LPS failures, such as system crashes. For instructions on these subjects, consult *LPS Operations and Maintenance Manual* and *LPS Installation Procedure* (References 3 and 4, respectively).

9.2 Backing Up the LPS Database

For Release 2, the LPS database should be manually backed up every day with an IRIX script, export_scr. Future releases should have an automatic backup. The script exists in the ORACLE login directory, /ora_backup (Figure 9–1).

```
export_scr script:
exp parfile=parfile_scr
-------
parfile_scr parameters:
full=y
file=export.dmp
grants=y
indexes=y
log=export.log
```

Figure 9–1. Database Backup Script

9.3 Purging the LPS Database

The LPS database contains records describing each contact, each L0R processing run, and each output file's transfer to EDC DAAC. The LPS software does not automatically delete these records. This allows the user to save Q&A information indefinitely, but also requires periodic removal of unneeded records. The current record retention period is 30 days. The SQL script mac_dbpurge in Figure 9–2 can be used to remove records associated with a contact captured before a date that the user specifies.

```
# lps_dbpurge - purge the LPS database of all Q&A records older than specified date

#   Usage: lps_dbpurge MM/DD/YY

#  Verifies that date argument is present;

#  Prompts for user name & password on stdout;

#  Starts up sqlplus and does the purge;

#  Purge is a single transaction--nothing is deleted if any delete fails.

#  WARNING: Date must be in MM/DD/YY format--correctness is not verified.

###############################################################

# Verify that the date (or something at least) is there

if [ $# -ne 1 ]

then

  echo "Usage: $0 MM/DD/YY"

  exit 1

fi

specdate=$1

# Accept user name and password

echo -n "User name: "

read up

if [ $up != "/" ]

then

  echo -n "Password: "

  stty -echo

  read pword

  stty echo

  echo ""

  up=$up/$pword

fi

sqlplus -S $up <<-ENDQS

set autocommit off;

REM RDC_Acct has the delete cascade option. All the records of the dependent

REM tables (LPS QA account tables) whose keys reference Contact_Sequence_Id

REM of RDC_Acct are deleted also.

delete from rdc_acct where

   actual_stop_time < to_date('$specdate', 'MM/DD/YY');

REM Valid_CCSDS_Parms is both a static and a history table

REM whose CCSDS_Parms_Id is referenced by RDP_Acct. Since it is also

REM a static table, there must exist a record in the table.

delete from valid_ccsds_parms

   where ccsds_Parms_Id not in (select unique ccsds_parms_id from rdp_acct)

     and insertion_time != (select max(insertion_time) from valid_ccsds_parms);

commit;

exit;

ENDQS
```

Figure 9–2. LPS Database Purge Script

To use the script to purge the LPS database, from the command line give the command lps_dbpurge MM/DD/YY, where MM/DD/YY is the date before which Q&A records are purged.

Executing the script requires select and delete permission on the LPS database tables for the LPS database instance.

9.4 Purging the LPS Journal

The LPS Journal file grows as messages are regularly added. The file can consume excessive amounts of disk space. Displaying the LPS Journal for browsing will require unacceptably long times to format the file for display. Therefore, periodically purge the LPS Journal file by deleting all of its contents and starting with a new empty file.

<div style="text-align:center">

WARNING

</div>

> This operation deletes the contents of the LPS Journal file. If you wish to save the LPS Journal file contents for future reference, make a copy of the file before purging it.

To delete the contents of the LPS Journal, type the following command to an IRIX shell:

```
% rm LPS_Journal
```

To make an empty LPS Journal file when none exists, type the following command to an IRIX shell:

```
% touch `grep local0.debug /etc/syslog.conf | awk '{print $NF}'`
```

This command requires write permission on the LPS Journal file.

To reinitiate the syslogd daemon, issue the command

```
killall – HUP syslogd
```

<div style="text-align:center">

NOTE

</div>

> Each LPS string has its own Journal file. To purge all Journal files, you must issue the touch command on each LPS string. The file is owned by root, and the `killall` command should be performed by the root.

9.5 Purging Trouble Files

Trouble files produced during L0R processing are not automatically deleted. They will consume excessively large amounts of disk space if allowed to accumulate. Therefore, periodically purge the directories containing trouble files.

```
┌─────────────────┐
│     WARNING     │
└─────────────────┘
```

This operation deletes trouble files. If you wish to save them for future reference, copy them to tape before purging the directories.

To purge the directories containing trouble files, type the following command to an IRIX shell:

```
% rm $LPS_TROUBLEFILE_PATH/*TroubleFile*
```

This command requires write permission on all trouble files.

NOTE

Each LPS string has its own directory for storing trouble files. To purge all trouble files, you must issue the `%  rm $LPS_TROUBLEFILE_PATH/*TroubleFile*` command on each LPS string.

9.6 Purging Saved LPS/EDC DAAC Communications Messages

Files containing saved messages between the LPS and EDC DAAC are not automatically deleted. If saving LPS/EDC DAAC communications is enabled, these files may consume excessively large amounts of disk space if allowed to accumulate. Therefore, periodically purge the directories containing saved communications messages.

```
┌─────────────────┐
│     WARNING     │
└─────────────────┘
```

This operation deletes the files. If you wish to save them for future reference, copy them to tape before purging the directories.

To purge the directories containing communications messages, type the following commands to an IRIX shell:

```
% rm $LPS_DANFILE_PATH/DAN_*MM.msg

% rm $LPS_DANFILE_PATH/DAA_*MM.msg

% rm $LPS_DDNFILE_PATH/DDN_*MM.msg

% rm $LPS_DDNFILE_PATH/DDA_*MM.msg
```

This command requires write permission on all LPS/EDC DAAC saved messages files.

NOTE

Each LPS string has its own directory for storing LPS/EDC DAAC messages. To purge all LPS/EDC DAAC message files, you must issue the commands above on each LPS string.

9.7 Purging the Reports Directory

NOT IMPLEMENTED IN RELEASE 3

9.8 File Consistency

File consistency is whether the files that exist on the system are also acknowledged as existing in the LPS database.

9.8.1 Raw Data Files

Only file consistency of raw data files can be checked at this time. At system startup time, check the raw data files with the database. If the raw data files exist in the system, the database should confirm this. The rdc_UpdAcct process searches for capture accounting information files and updates the database.

9.8.2 Output Data Files

NOT IMPLEMENTED IN RELEASE 3

10. LPS GUI Detailed Reference

10.1 Introduction

This chapter explains each LPS GUI command. Explanations are arranged according to the main menu's top-level pulldown menus. Figure 10–1 shows a view of the LPS main menu bar. (All the figures are located at the end of the chapter.) Each explanation includes a description of what the menu option does, what additional information you may need to provide, what default values are used when you do not supply additional information, and explanations of any additional dialogs displayed by the LPS GUI.

This chapter does not provide step-by-step explanations for performing LPS procedures. To find out how to carry out a specific procedure, refer to Sections 3 through 9.

10.2 Setup Menu

The Setup menu (Figure 10–2) contains commands to view and edit information on LPS string configuration, L0R processing parameters, and L0R processing error reporting thresholds.

10.2.1 View/Edit Capture Source...

Selecting this option brings up the Set Capture Source dialog (Figure 10–3). Information about the capture source values is described in Table 10–1. Clicking CANCEL resets the capture source to the original value shown. Clicking OK will update the database accordingly.

10.2.2 Ingest Contact Schedules...

Selecting this option brings up the Ingest Contact Schedules dialog (Figure 10–4). Clicking CANCEL will leave the dialog with no actions taken. Clicking OK will ingest the selected contact schedule and propagate it to the selected strings (or instance in the case of string 5).

10.2.3 View/Edit Contact Schedule...

Selecting this option brings up the Edit Contact Schedules dialog (Figure 10–5). The contact schedule for the current string will be viewed. The fields of the schedule may be edited as desired. Clicking CANCEL will leave the dialog with no actions taken. Clicking OK will make the desired changes to the contact schedule for that string. Clicking INSERT will add a blank line for manual addition of contact information. Clicking DELETE will bring up a standard confirmation dialog.

10.2.4 Load IAS Parameter File...

Selecting this option brings up the Load IAS Parameter File dialog (Figure 10–6). Clicking OK will load in the selected IAS CPF. Clicking CANCEL will leave the dialog with no actions taken.

Table 10–1. Additional Information for the Start Capture Command

Item	Description
Capture Source	The source of the data to be captured. The default is the current capture source in the LPS string's configuration. You can select an alternate source by holding down the mouse button over the field and selecting one of the options. The options and their meanings are as follows: • 1I = X-band 1 I channel • 1Q = X-band 1 Q channel • 2I = X-band 2 I channel • 2Q = X-band 2 Q channel • 3I = X-band 3 I channel • 3Q = X-band 3 Q channel • TB = LGS bit error rate tester • T1 = Test data from LPS string 1 • T2 = Test data from LPS string 2 • T3 = Test data from LPS string 3 • T4 = Test data from LPS string 4 • T5 = Test data from LPS backup/development string
Isolate Capture Process	Dedicate a processor to perform this operation.
Suspend L0R Processing	Suspend L0R processing on this string during this operation.
Duration	The duration of time after which data capture should stop. The default is 14 minutes – the maximum possible duration of a single contact. You may enter another time by typing the value (format = min:sec) into the field. The duration is measured from the point of first data capture.
Station of Origin	Select the station where the data was collected from the satellite. The default is EDC. For data received on tape at LGS from supplementary ground stations (e.g., AGS or SGS), select the appropriate station.

10.2.5 Propagate L0R Parameters...

Selecting this option brings up the Propagate Level 0R Parameters dialog (Figure 10–7). Clicking CANCEL will leave the dialog with no actions taken. Clicking OK will propagate the L0R parameters to the selected strings.

10.2.6 View/Edit L0R Parameters...

Selecting this option brings up the Edit Level 0R Parameters dialog (Figure 10–8). Current values for the L0R parameters are displayed, with some values needing individual scrolling to be seen. Inserting invalid values for the parameters will bring a message at the bottom of the window notifying the user of the valid range of the field in question. Clicking CANCEL will leave the dialog with no actions taken. Clicking OK will update the parameters on the string in use.

10.2.7 View/Edit L0R Thresholds...

Selecting this option brings up the Edit Level 0R Thresholds dialog (Figure 10–9). Current values for the L0R error thresholds are displayed, with some values needing individual scrolling to be seen. Inserting invalid values for the parameters will bring a message at the bottom of the window notifying the user of the valid range of the field in question. Clicking CANCEL will leave the dialog with no actions taken. Clicking OK will update the parameters on the string in use.

10.2.8 View/Edit Output File Transfer Config...

Selecting this option brings up the View/Edit Output File Transfer Config dialog (Figure 10–10). The current parameters for transferring files to the ECS are displayed. Clicking CANCEL will leave the dialog with no actions taken. Clicking OK will update the parameters on the string in use.

10.3 Control Menu

The Control menu (Figure 10–11) contains commands to perform LPS functions such as data capture, copying to tape, and L0R processing, as well as commands to enable or disable LPS automatic processes.

10.3.1 Start Capture...

Selecting this option causes the LPS software to begin capturing raw wideband data. The LPS GUI displays the Start Data Capture dialog (Figure 10–12), requesting further information about the contact or test data transmission to be captured. Table 10–1 contains capture source and other information. Clicking CANCEL cancels the operation. Clicking OK starts data capture immediately using the information the user has provided in the dialog.

10.3.2 Stop Capture...

Selecting this option brings up the Stop Data Capture Confirmation dialog (Figure 10–13). Clicking YES causes the current capturing to be halted. Clicking NO cancels the operation; LPS software continues to capture data.

10.3.3 Start L0R Processing...

Selecting this option causes the LPS software to begin processing a file containing raw wideband data to L0R. The LPS GUI displays the Start Level 0R Processing dialog (Figure 10–14), requesting further information about the contact or test data transmission to be processed as described in Table 10–2. Clicking CANCEL cancels the operation. Clicking OK starts L0R processing immediately using the information the user has provided in the dialog.

Table 10–2. Additional Information for the Start Level 0R Processing Command

Item	Description
Raw Data File	The file containing raw wideband data to be processed. The dialog displays identifying information for each raw wideband data file stored on the string's raw data storage array. Select the file by clicking on any of the fields describing it. The selected file is highlighted. When the dialog is first displayed, the first file in the list is selected by default.
File Version Number	The right-most field for each raw data file contains the file version number that will be assigned to this instance of L0R processing for this file. By default, the value is one greater than the highest value recorded for this raw wideband data file in the string's database. Enter another value if, for instance, this raw data has already been processed with this file version number on another string. NOTE: If the file version number supplied is greater than the default, the LPS GUI will issue a warning dialog asking the user to confirm the selection. Once confirmed, the LPS software will use the specified file version number. The LPS software will not accept a file version number that has already been used for another instance of L0R processing on this file on this string.
Delete the raw data file after processing	When selected, this option will cause the raw data file to be deleted automatically from the string's raw data storage array after it has been successfully processed. The default is NOT to delete the file. NOTE: The file will be deleted only if both L0R processing completes successfully and the raw data file has been copied to the 30-day store. Otherwise, the file will remain online.
Delete the output files after transmission	When selected, this option will cause the output files to be deleted automatically from the string's output file storage array after they are successfully transferred to EDC DAAC. The default is to delete the files automatically.

10.3.4   Stop L0R Processing...

Selecting this option causes the LPS software to stop a specified instance of L0R processing. The LPS GUI displays the Stop Level 0R Processing dialog (Figure 10–15), displaying a list of the raw data files being processed along with the file version number assigned to this instance of its processing. Select the instance of L0R processing to terminate. The selected line is highlighted. By default, the first line is selected. Clicking CANCEL cancels the operation. Clicking OK causes the LPS GUI to display the confirmation dialog. Clicking OK a second time terminates the selected instance of L0R processing. Clicking CANCEL cancels the operation and returns the user to the Stop Level 0R Processing dialog.

This option is enabled only when the LPS software is currently processing at least one raw wideband data file to L0R.

10.3.5   Start Copy to Tape...

Selecting this option brings up the Start Copy to Tape dialog (Figure 10–16). The available raw data files are displayed, along with contact start and stop times. Only one file may be selected for copying. Clicking OK will copy the selected raw data file to tape. Clicking CANCEL will leave the dialog with no actions taken. The delete raw file option will delete the raw file on completion of the copy action.

10.3.6   Stop Copy to Tape...

When this option is selected, the LPS GUI will display a standard confirmation dialog (Figure 10–17). Clicking OK causes the current copying action to be halted. Clicking CANCEL cancels the operation; the LPS software continues to copy data to tape.

10.3.7   Start Restage...

Selecting this option will allow the LPS to restore raw data from tape to disk. The LPS GUI will display the Start Restage dialog (Figure 10–18) to confirm the action. Clicking OK will start the restage operation. Clicking CANCEL will leave the dialog with no actions taken.

10.3.8   Stop Restage...

When this option is selected, the LPS GUI will display Stop Restage dialog (Figure 10–19) to quiz the user on this course of action. This dialog looks similar to the Start Restage dialog, except with the word Stop instead of Start. Clicking OK causes the current restage action to be halted. Clicking CANCEL cancels this operation; the LPS software continues to restore raw data from tape to disk.

10.3.9   Start Auto Capture...

Selecting this option causes the LPS software to begin the auto data capture process that will capture data, perform L0R processing of the data, and archive raw data to tape automatically. The LPS GUI displays the Start Auto Data Capture dialog (Figure 10–20) to confirm this course of action. Clicking OK will start the auto data capture operation. Clicking CANCEL will leave the dialog with no actions taken.

10.3.10   Stop Auto Capture...

When this option is selected, the LPS GUI will display Stop Auto Capture dialog (Figure 10–21) to quiz the user on this course of action. This dialog looks similar to the Start Auto Capture dialog, except with the word Stop instead of Start. Clicking OK causes the current auto capture to be halted. Clicking CANCEL cancels this operation; the LPS software continues automatic data capture.

10.3.11   Generate Tape Label...

Selecting this option brings up the Generate Tape Label dialog (Figure 10–22). Selected raw data files will be displayed. One file may be selected to have a tape label generated. Clicking OK will start generating the tape label. Clicking CANCEL will leave the dialog with no actions taken.

10.4 Reports Menu

The Reports menu (Figure 10–23) contains commands to generate LPS data receive, LPS Q&A, periodic Q&A, and file transfer summary reports. The **PRINT** option on each of the reports can generate a postscript file is desired.

10.4.1 Data Receive Summary...

Selecting this option brings up the Data Receive Summary Report dialog (Figure 10–24). A selection of contact periods will be displayed. Clicking OK will generate a report concerning the selected contact period. Clicking CANCEL will leave the dialog with no actions taken.

10.4.2 LPS Quality/Accounting...

Selecting this option brings up the LPS Quality/Accounting Report dialog (Figure 10–25). A selection of contact periods and file version numbers will be displayed. Clicking OK will generate a report concerning the selected contact period/file version number. Clicking CANCEL will leave the dialog with no actions taken.

10.4.3 Periodic Quality/Accounting...

Selecting this option brings up the LPS Periodic Quality/Accounting Report dialog (Figure 10–26). A selection of contact periods and file version numbers will be displayed. Clicking OK will generate a report for the selected contact period/file version number. Clicking CANCEL will leave the dialog with no actions taken.

10.4.4 File Transfer Summary...

Selecting this option brings up the Data Transfer Summary Report dialog (Figure 10–27). The user will need to supply a begin and end time for the transfer report. Clicking OK will generate a report concerning the period in question. Files transferred from data captured in the period will be reported. Clicking CANCEL will leave the dialog with no actions taken.

10.5 Test Menu

The Test menu (Figure 10–28) contains a single command to play back a captured raw wideband data file.

10.5.1 Send Data...

Selecting this option brings up the Send Data dialog (Figure 10–29). A selection of available raw data files will be displayed. Information about the Isolate Processor and Suspend L0R Processing

options is described in Table 10–2. Clicking OK will send the selected data. Clicking CANCEL will leave the dialog with no actions taken.

10.6 Monitor Menu

The Monitor menu (Figure 10–30) contains commands to view LPS software status and error messages, either by displaying messages as they are generated or by displaying the messages stored in the LPS Journal file.

10.6.1 Add LPS Journal Entry...

Selecting this option brings up the LPS Journal Entry window (Figure 10–31), where the user enters a text string. Clicking OK will cause the text string to be written to the LPS Journal file. Clicking CANCEL will leave the window with no action taken.

10.6.2 Display LPS Journal File...

Selecting this option causes the LPS software to create a window (Figure 10–32) displaying the formatted contents of the LPS Journal file. This window contains menus allowing the user to customize the display format; filter the messages displayed by priority or other characteristics; sort the messages by priority or time; print, mail, or save selected messages; and perform other functions. Online help for the window's options is available through the window's **Help** menu. Additional information is available from sysmon(1) in Reference 2.

NOTE

The LPS Journal file display window maps the eight LPS message priorities to four categories of priority. See Section 2.7.2 for details.

To terminate the display, select **File>Quit** from the window's menu bar.

10.6.3 Display Operational Messages...

Selecting this option causes the LPS software to create a window in which LPS status and error messages will be displayed as they are generated. The LPS GUI displays the Display Operational Messages dialog (Figure 10–33), requesting the user to select a set of message priorities for display. The message display window will display only the priorities selected. By default, all message priorities except Debug are selected for display (see Table 2–1 for an explanation of LPS message priorities and their meanings). Clicking CANCEL cancels the operation; no display is created. Clicking OK creates a display window (Figure 10–34) for messages of the selected priorities. The window can be moved, resized, or iconified and the display font can be changed. For details, see Chapter 8, "Managing Windows," in *IRIX Essentials* (Reference 5) and the xwsh(1G) in Reference 2.

To terminate the display, move the mouse to the window's menu bar, hold down the right mouse button, and select **Quit** from the pulldown menu.

10.7 File Management Menu

The File Management menu (Figure 10–35) contains commands related to LPS output file handling. Commands in this menu allow the user to send a DAN to EDC DAAC, indicate that a given set of output files should be retained or deleted automatically after successful transfer to EDC DAAC, delete an output file set immediately, and enable/disable the automatic receipt of DDNs.

10.7.1 DAN Transfer State...

Selecting this option brings up the Set DAN Transfer State dialog (Figure 10–36). This dialog toggles the ability for automatic notification to EDC DAAC of output file availability. Clicking OK will set the desired DAN transfer state. Clicking CANCEL will leave the dialog with no actions taken.

10.7.2 Resend Suspended DAN...

Selecting this option brings up the Resend Suspended DAN dialog (Figure 10–37). Clicking OK will resend the suspended DAN. Clicking CANCEL will leave the dialog with no actions taken.

10.7.3 Resend Failed DAN...

Selecting this option brings up the Resend Failed DAN dialog (Figure 10–38). The failed DANs are displayed, along with contact start and stop times. Clicking OK will resend the selected failed DAN. Clicking CANCEL will leave the dialog with no actions taken.

10.7.4 Output File Set Management...

Selecting this option brings up the Output File Set Management dialog (Figure 10–39). Output file sets are displayed with information about the capture string ID, contact times, file version number, transfer status, whether the files should or should not be retained after transfer, and whether the files should be deleted now. Clicking OK will perform the desired actions on the selected output file set. Clicking CANCEL will leave the dialog with no actions taken. Table 10–3 contains information about the retain after transfer and delete output file set buttons.

10.7.5 Delete Raw File...

Selecting this option brings up the Delete Raw Data File dialog (Figure 10–40). Displayed are the raw file names, the disk volume in megabits, whether the raw data file has been archived, and whether it has been processed. Clicking OK will delete the selected raw data file. Clicking CANCEL will leave the dialog with no actions taken.

10.7.6 Start DDN Server...

Selecting this option causes the LPS software to stop the automatic receipt and processing of DDNs from EDC DAAC. The LPS GUI displays the Stop DDN Server dialog (Figure 10–41) requesting the user to confirm the command. Clicking CANCEL cancels the operation; DDNs will be received and processed. Clicking OK terminates the DDN server process.

Table 10–3. Additional Information for the Output File Set Management Command

Item	Description
Retain after transfer	If this item is selected, the output files will be retained after transfer. An X will appear in the Retain After Transfer column. If an X is already in the column, this option may be unselected so that the output files are deleted after transfer.
Delete output file set	If this item is selected, a confirmation dialog will appear asking if you really want to delete the files. Click the OK button if this is desired. If no files have been transferred, an A will appear in the Delete Now column. If only some of the files have been transferred, another dialog appears asking to delete only the transferred files or all the files. If all is selected, an A will appear in the Delete Now column. If only the transferred files, a T will appear in the column.

10.7.7 Stop DDN Server...

When this option is selected, the LPS GUI displays the Start DDN Server dialog (Figure 10–42). Clicking CANCEL cancels the operation; DDNs will not be received and processed. Clicking YES activates the DDN server process. DDNs will be received and processed automatically.

10.8 Shutdown Menu

There is no pulldown menu for the Shutdown option. Clicking the mouse on this option will cause the LPS GUI to attempt to shut down the LPS software. The LPS GUI will display the Shutdown LPS dialog (Figure 10–43). Clicking CANCEL will cancel the operation, leaving the state of the LPS software unchanged. Clicking OK causes the LPS GUI to attempt the shutdown. The LPS GUI itself will exit and all LPS software automatic processes will be terminated.

If the LPS software is currently capturing data or performing L0R processing, the shutdown command will display the Shutdown Command Failure dialog (Figure 10–44). Clicking OK is the only option. In this case, the state of the LPS software is unchanged.
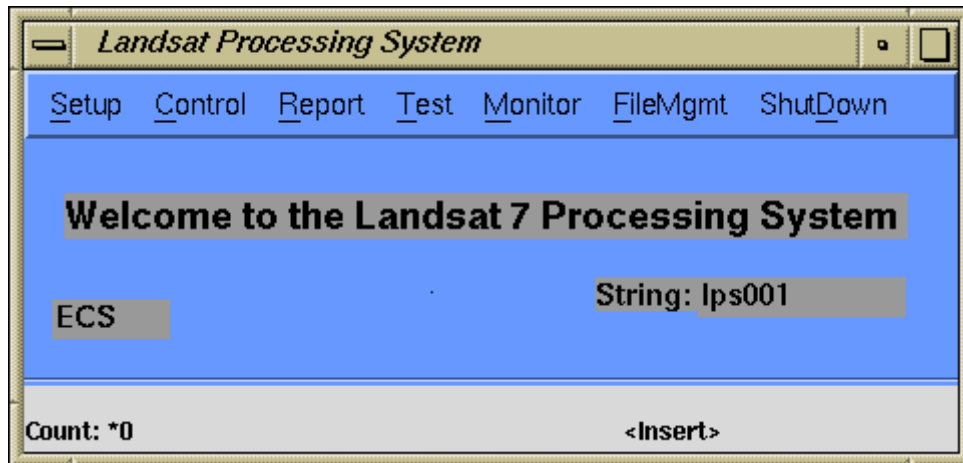
Figure 10–1. LPS GUI Main Menu
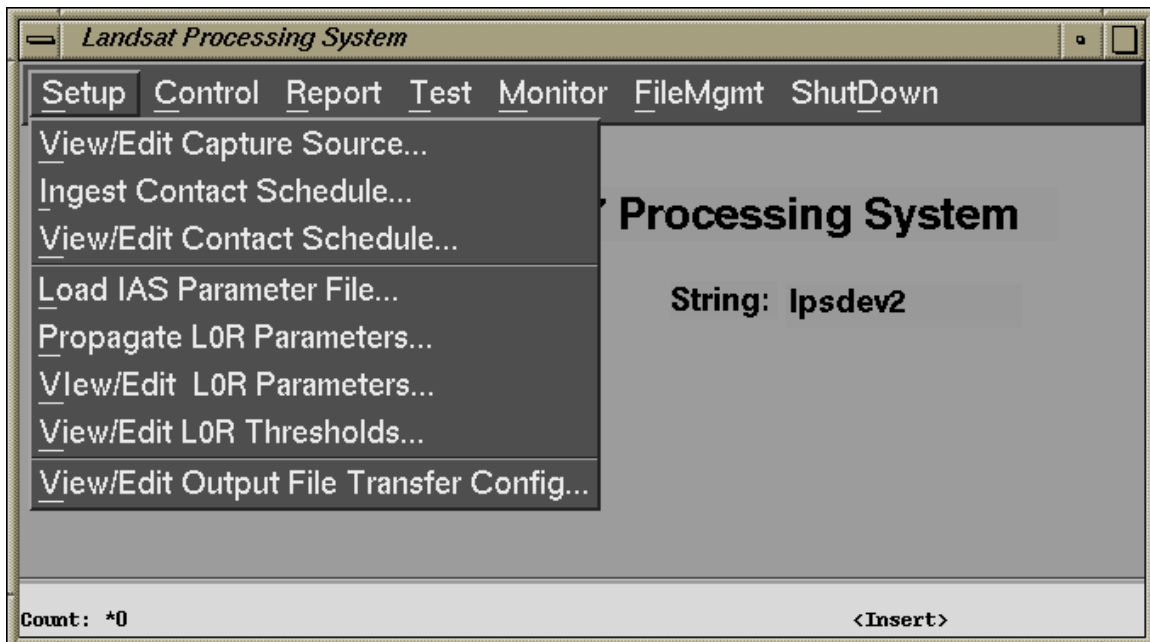
Figure 10–2. LPS GUI Setup Menu
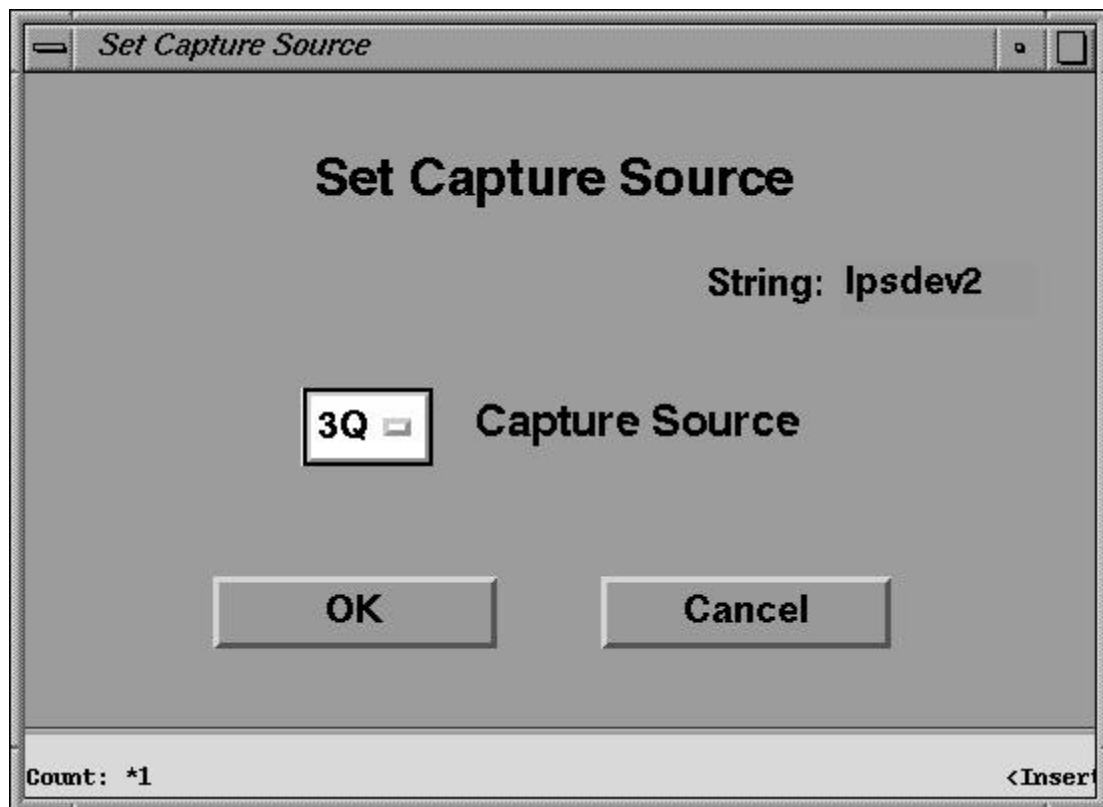
Figure 10–3. LPS GUI Set Capture Source Dialog
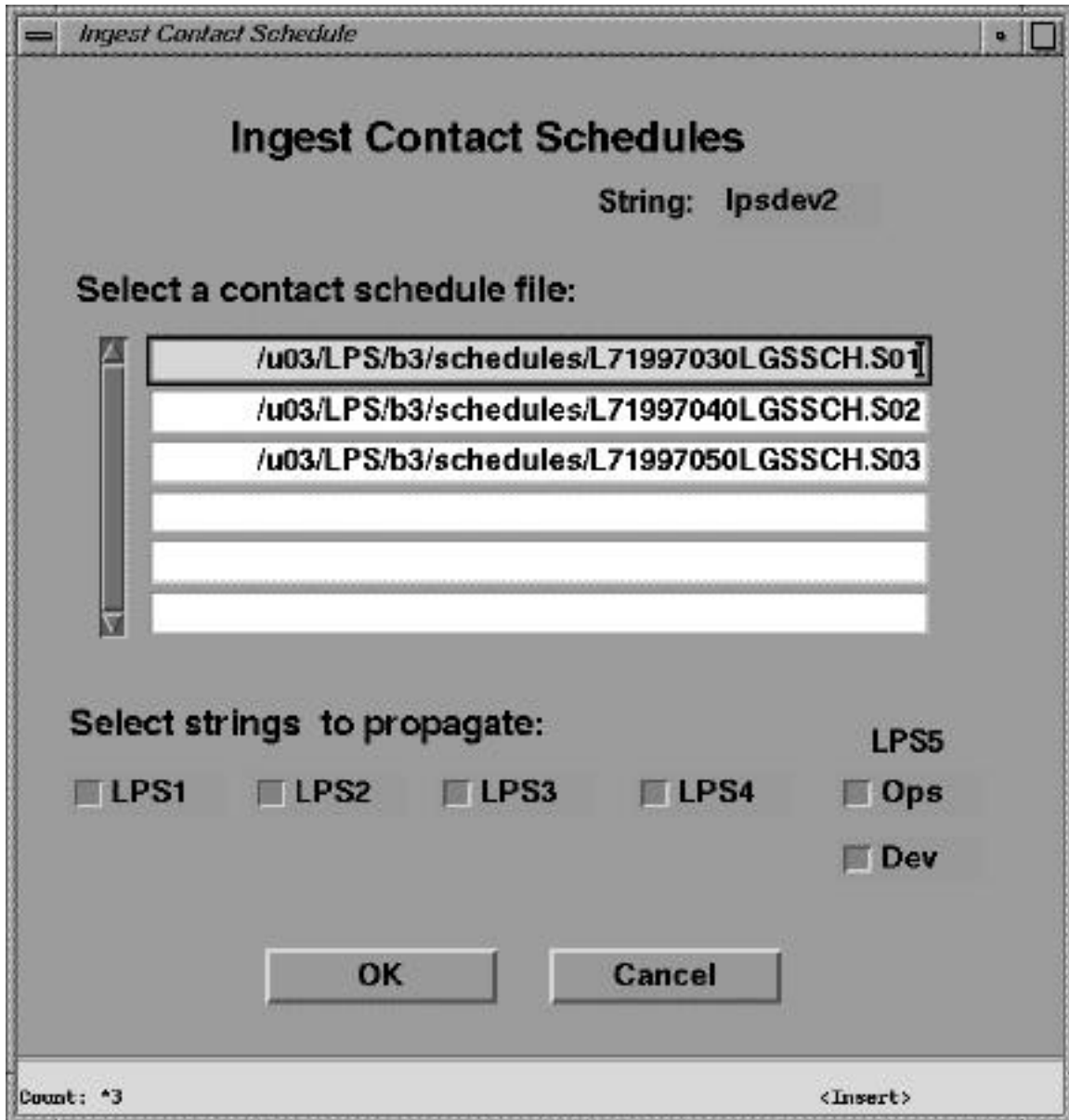
Figure 10–4. LPS GUI Ingest Contact Schedules Dialog

Edit Contact Schedules

String: lpsdev2

Scheduled Start Time	Scheduled Stop Time
17–MAY–1997–02:43:31	17–MAY–2000–02:49:30
17–MAY–1997–02:43:31	17–MAY–1997–02:52:42
17–MAY–1997–04:20:46	17–MAY–1997–04:25:09
17–MAY–1997–16:19:14	17–MAY–1997–16:24:01
17–MAY–1997–16:19:50	17–MAY–1997–16:28:13
17–MAY–1997–17:57:38	17–MAY–1997–18:06:00

Insert

Delete   OK   Cancel

Count: 6   v   <Insert>
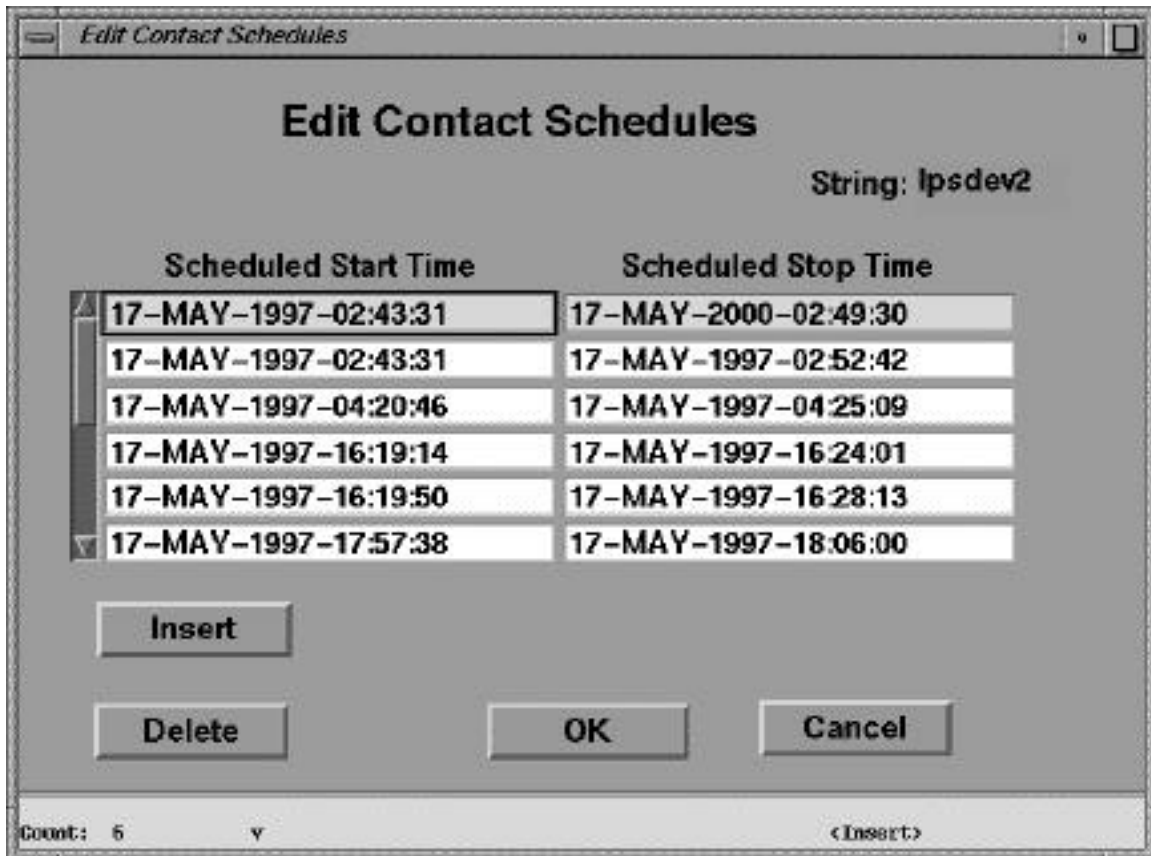
Figure 10–5. LPS GUI Edit Contact Schedules Dialog
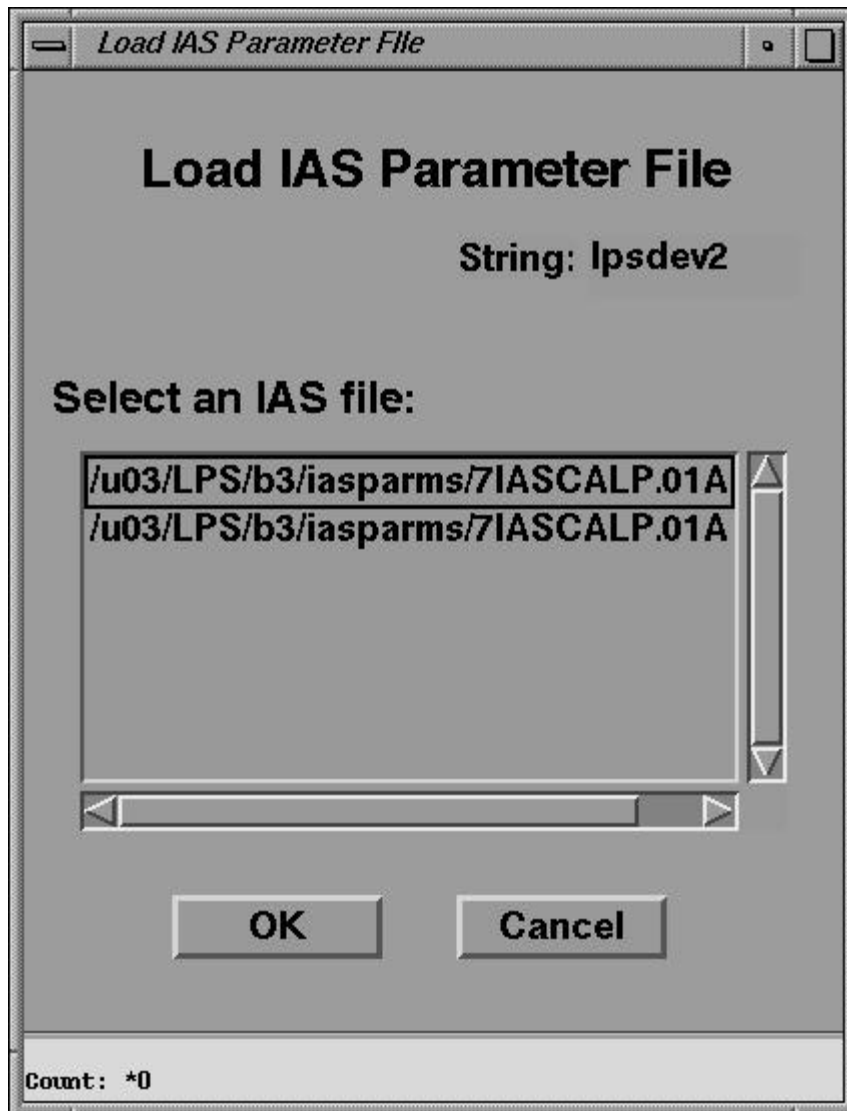
Figure 10–6. LPS GUI Load IAS Parameter File Dialog

Propagate Level 0R Parameters

String: lpsdev2

Select strings to propagate:

- [ ] LPS1
- [ ] LPS2
- [ ] LPS3
- [ ] LPS4
- [ ] LPS5

OK   Cancel

Count: *0
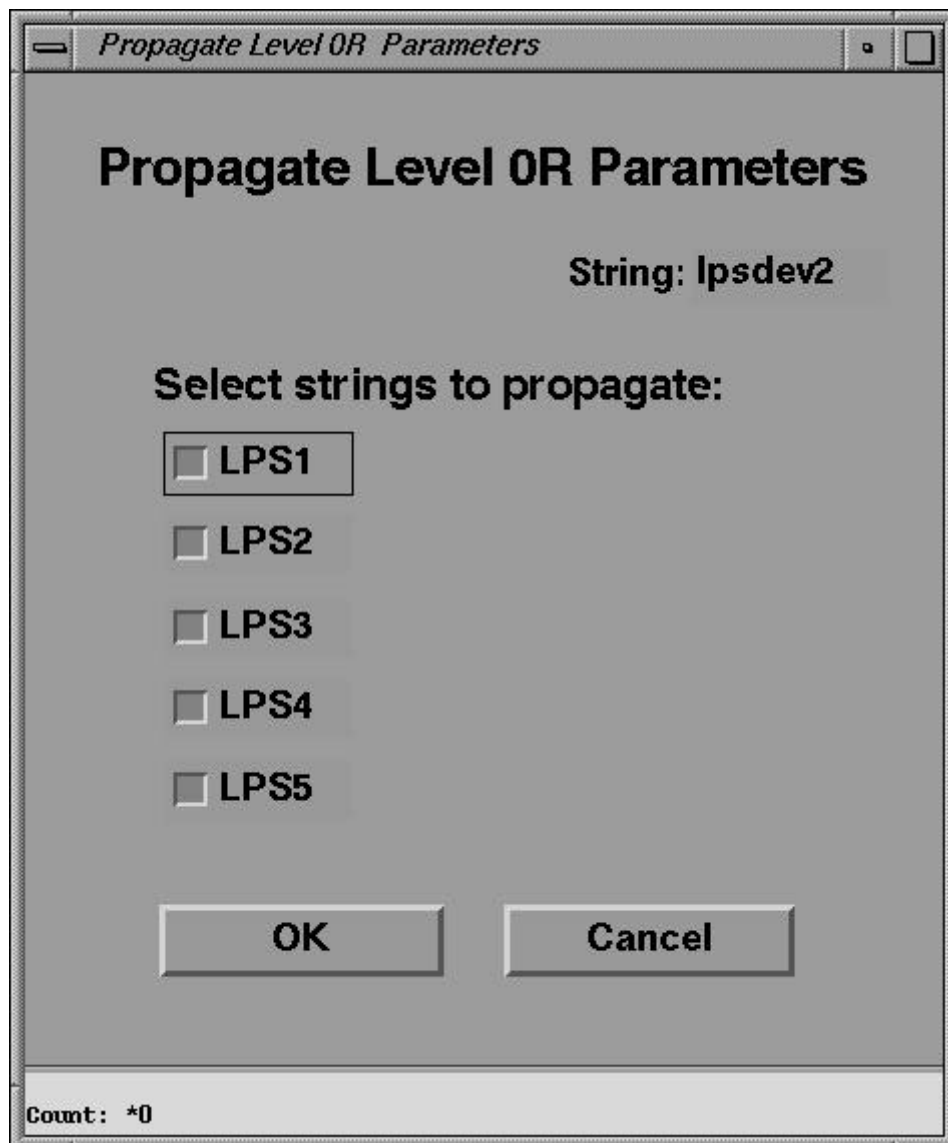
Figure 10–7. LPS GUI Propagate Level 0R Parameters Dialog

Figure 10–8. LPS GUI Edit Level 0R Parameters Dialog (1 of 2)

Figure 10–8. LPS GUI Edit Level 0R Parameters Dialog (2 of 2)

Figure 10–9. LPS GUI Edit Level 0R Thresholds Dialog

Figure 10–10. LPS GUI View/Edit Output File Transfer Config Dialog
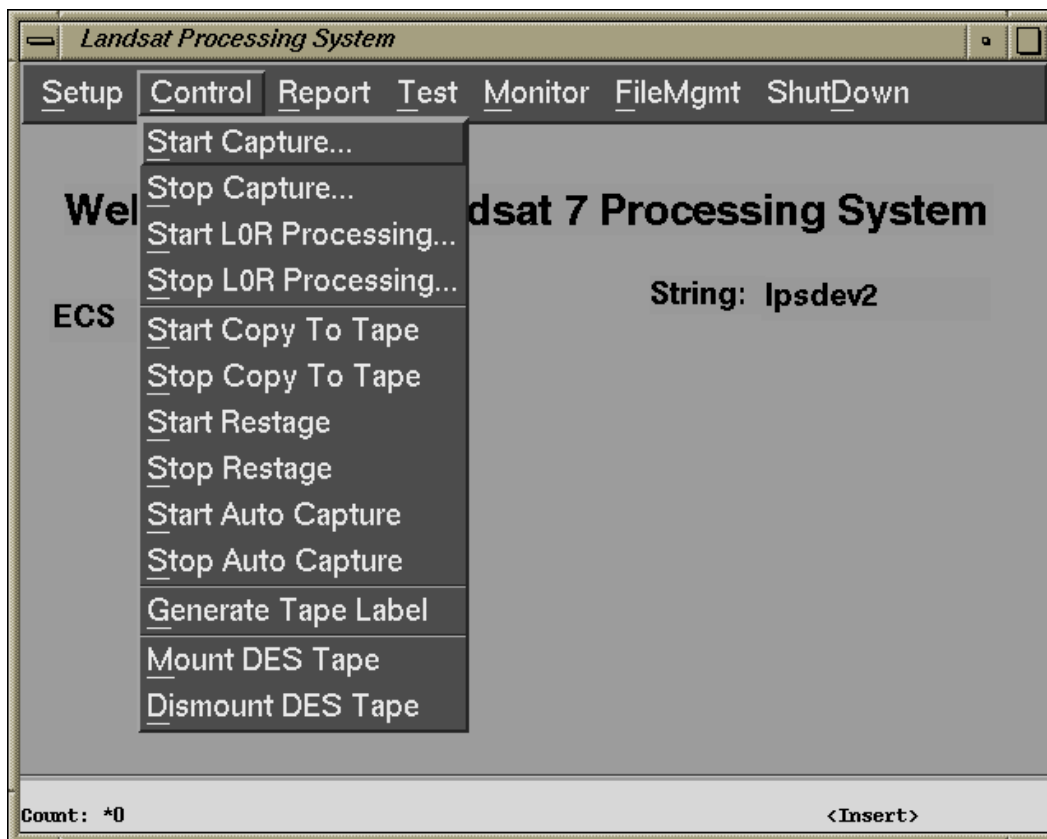
Figure 10–11. LPS GUI Control Menu
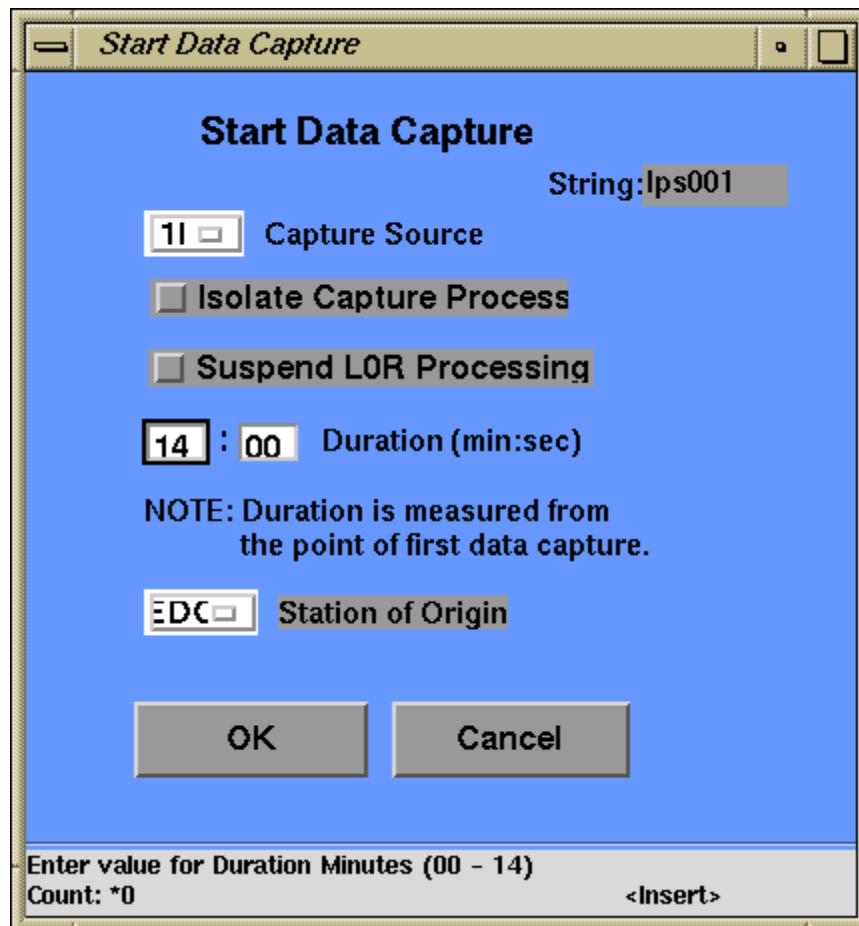
Figure 10–12. LPS GUI Start Data Capture Dialog
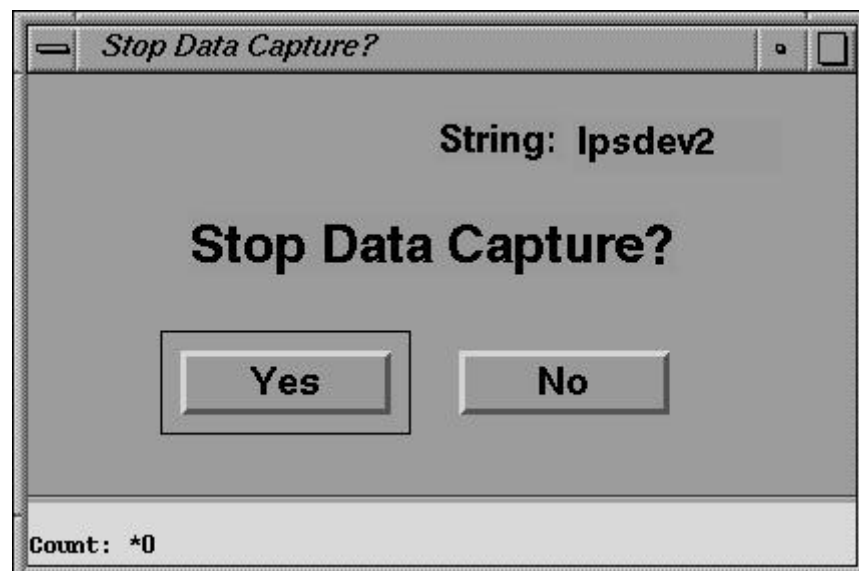
Figure 10–13. LPS GUI Stop Data Capture Confirmation Dialog
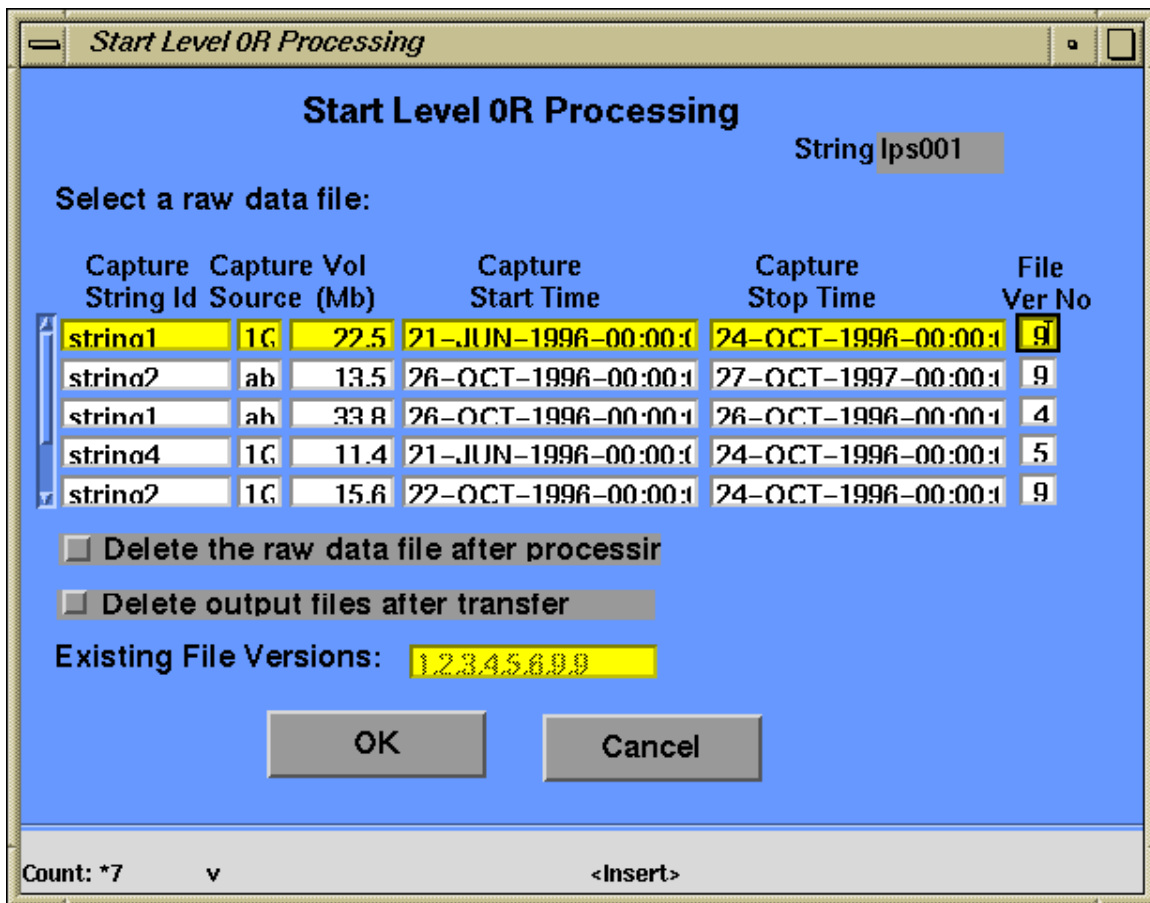
Figure 10–14. LPS GUI Start Level 0R Processing Dialog
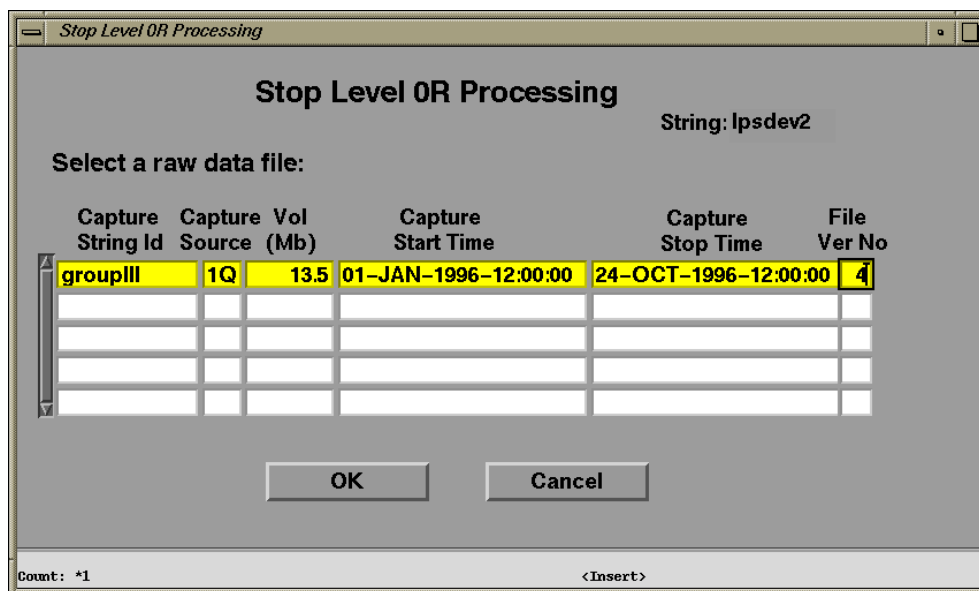
Figure 10–15. LPS GUI Stop Level 0R Processing Dialog
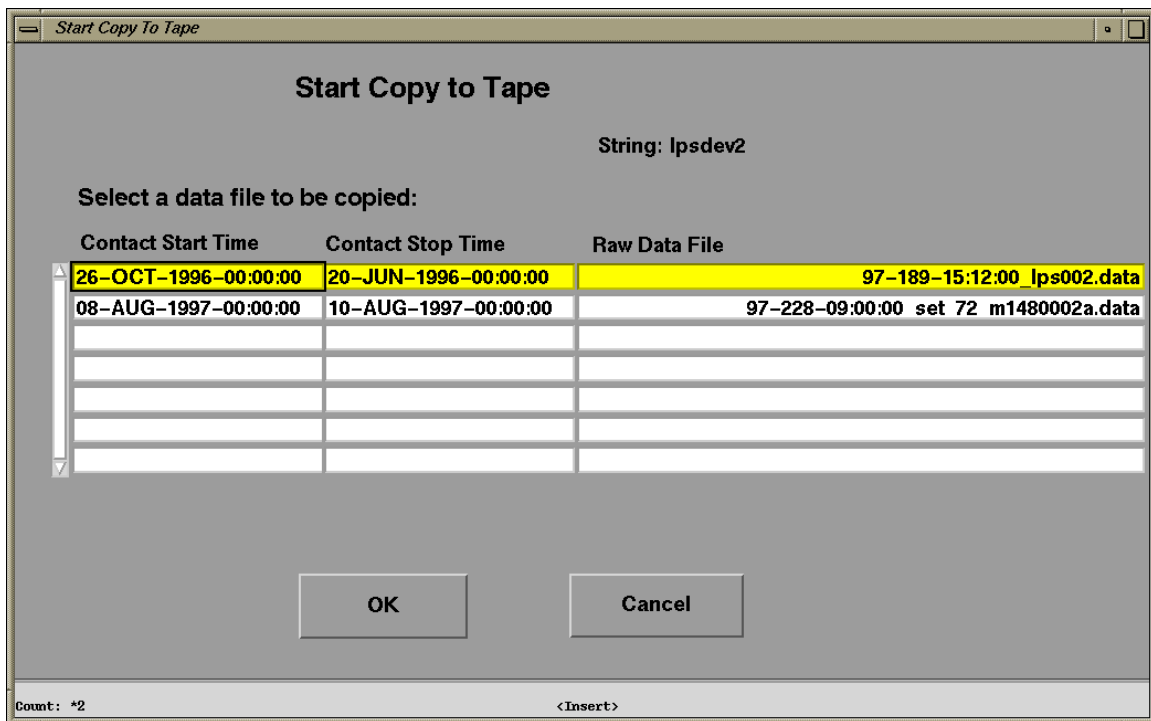
Figure 10–16. LPS GUI Start Copy to Tape Dialog

Figure 10–17. LPS GUI Stop Copy to Tape Confirmation Dialog
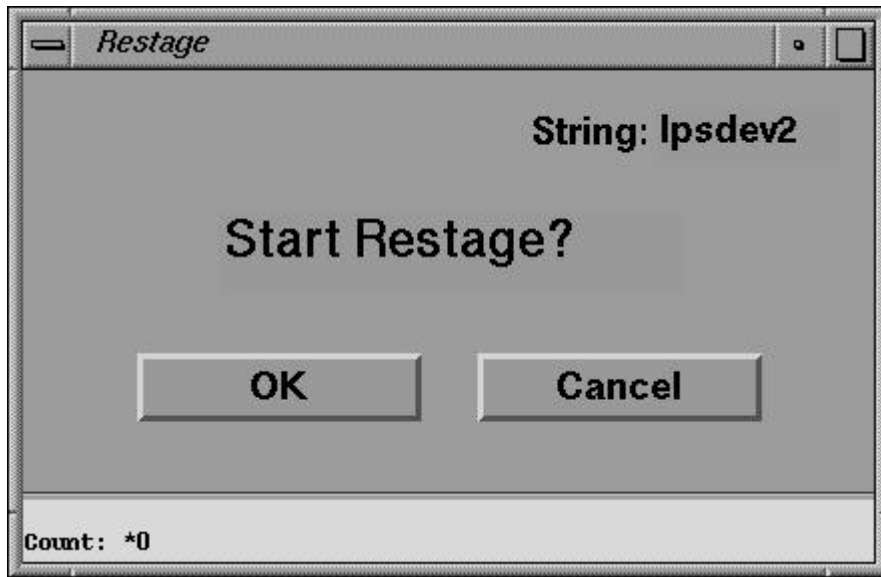
Figure 10–18. LPS GUI Start Restage Confirmation Dialog
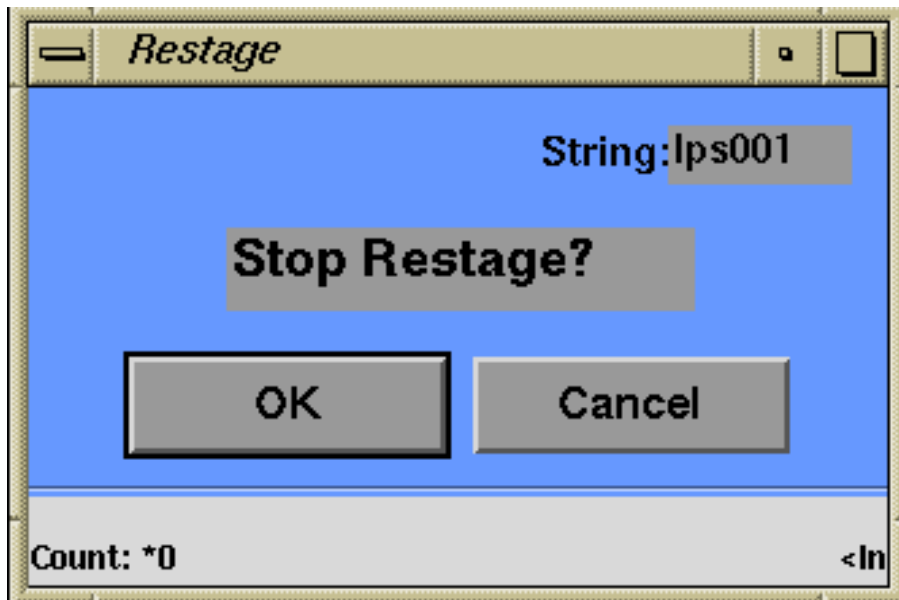
Figure 10–19. LPS GUI Stop Restage Confirmation Dialog
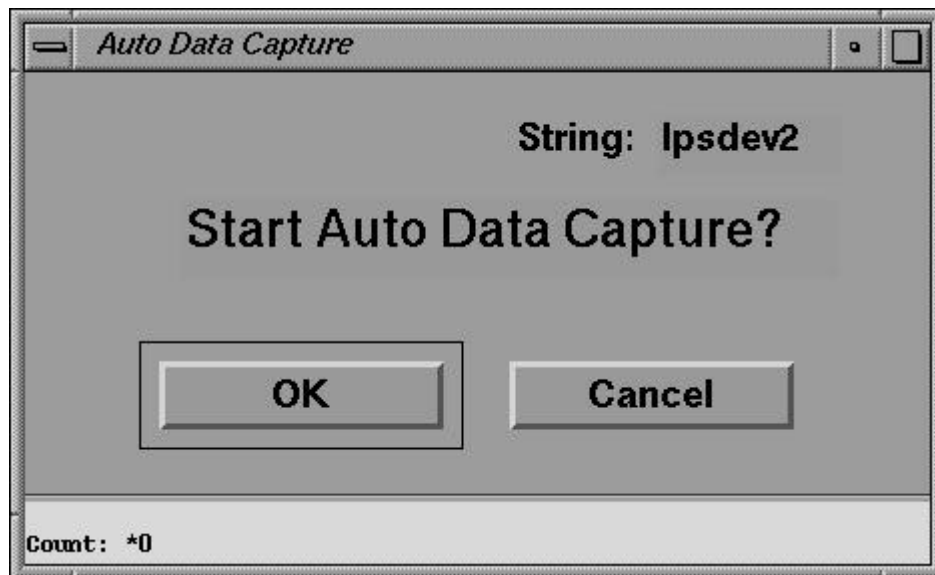
Figure 10–20. LPS GUI Start Auto Data Capture Confirmation Dialog
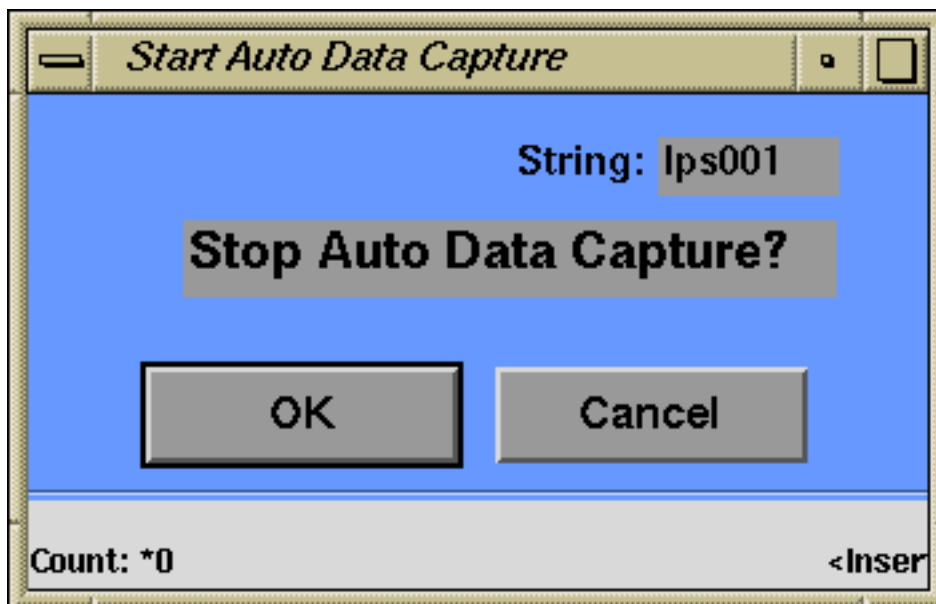
Figure 10–21. LPS GUI Stop Auto Data Capture Confirmation Dialog
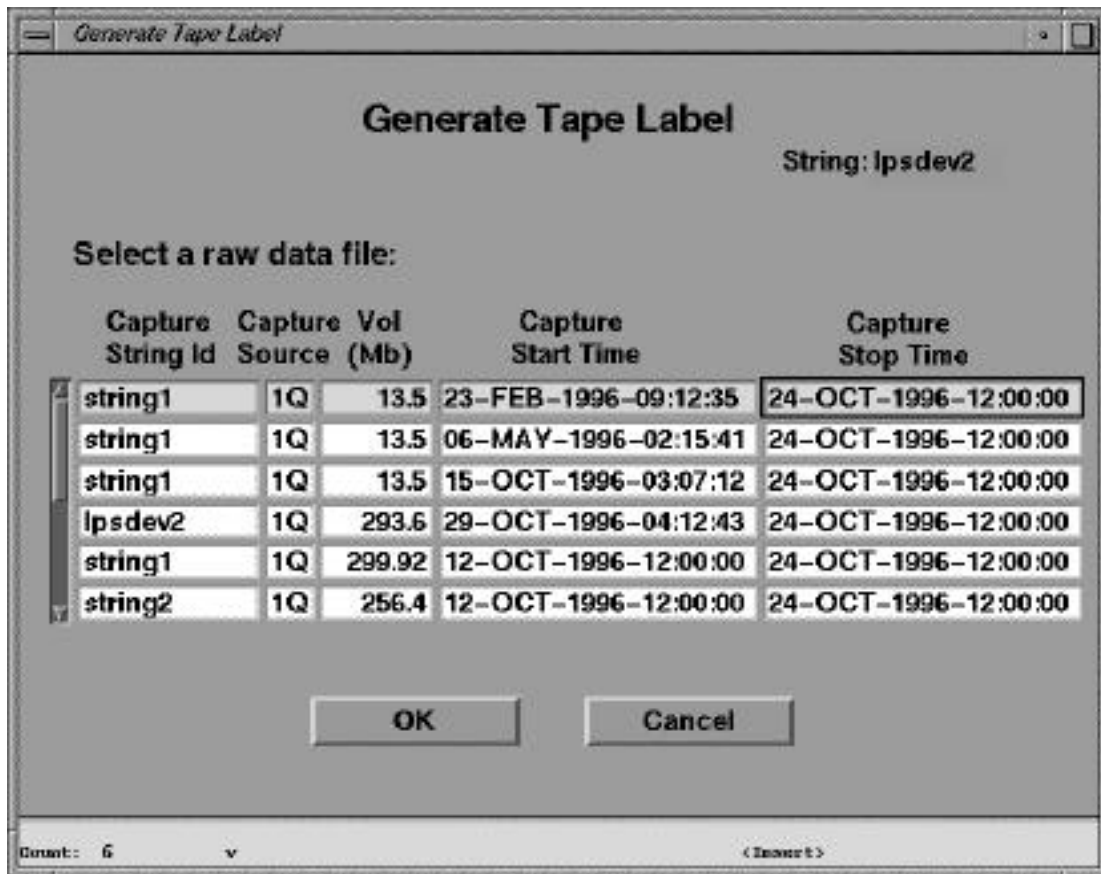
Figure 10–22. LPS GUI Generate Tape Label Dialog
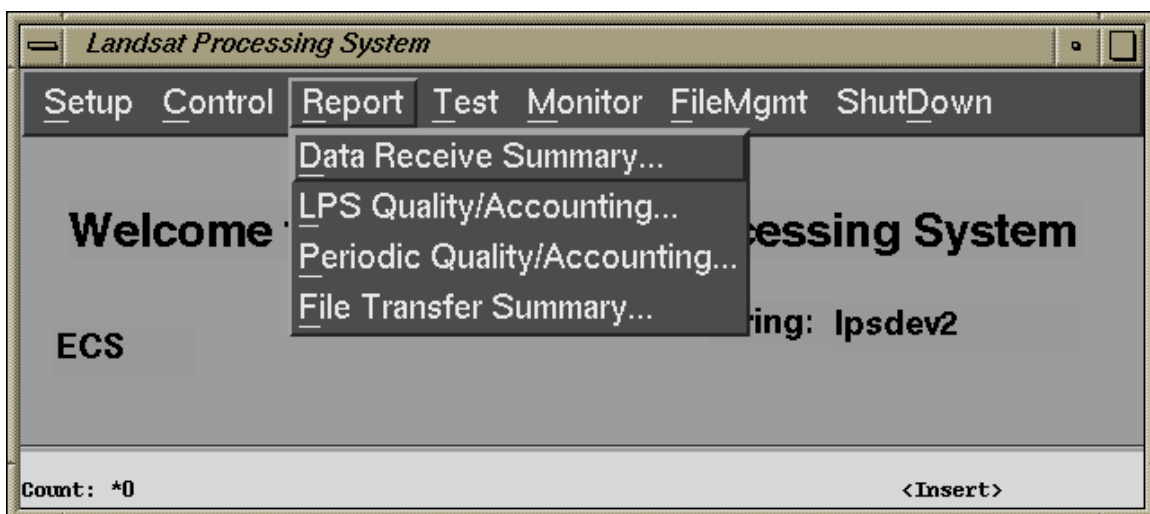
Figure 10–23. LPS GUI Reports Menu
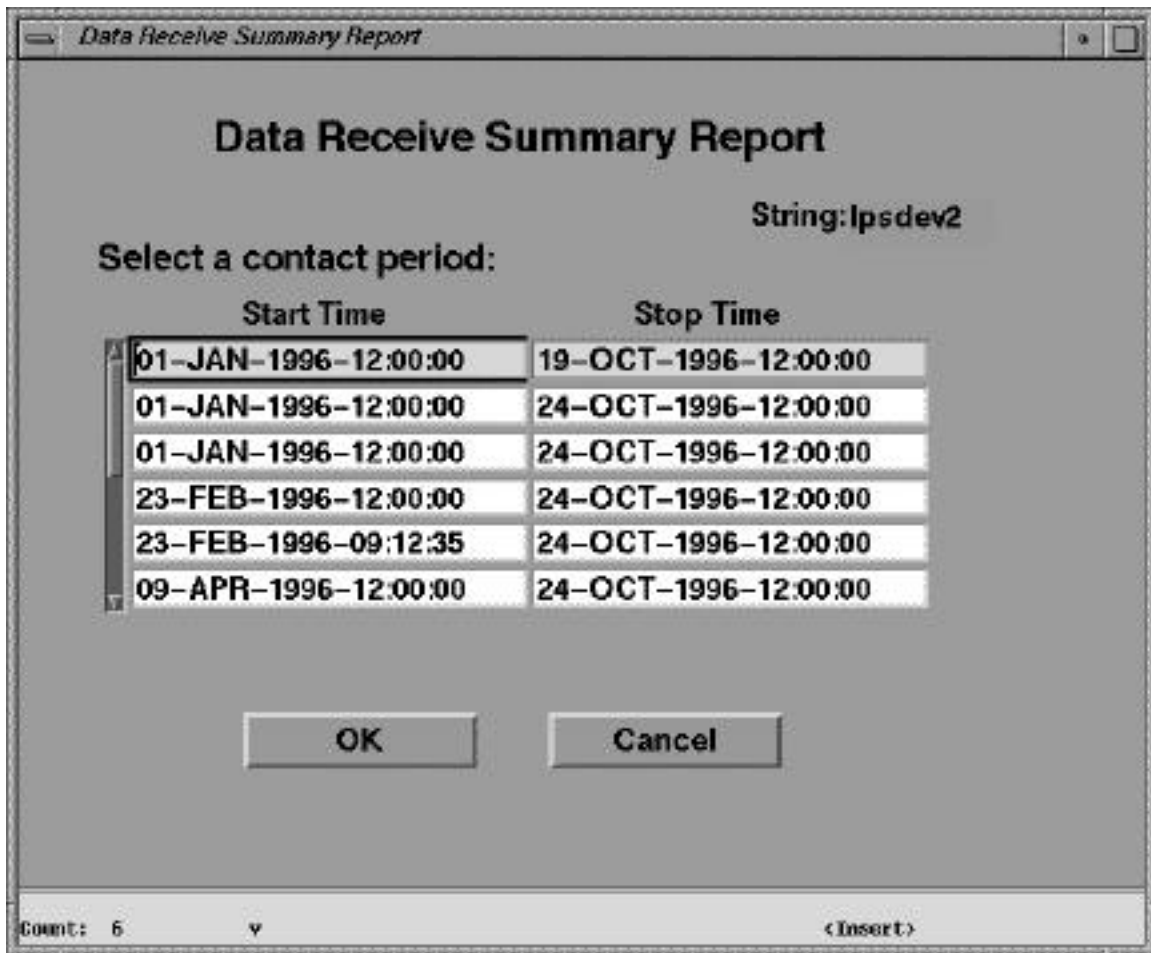
Figure 10–24. LPS GUI Data Receive Summary Report Dialog

Figure 10–25. LPS GUI LPS Quality/Accounting Report Dialog

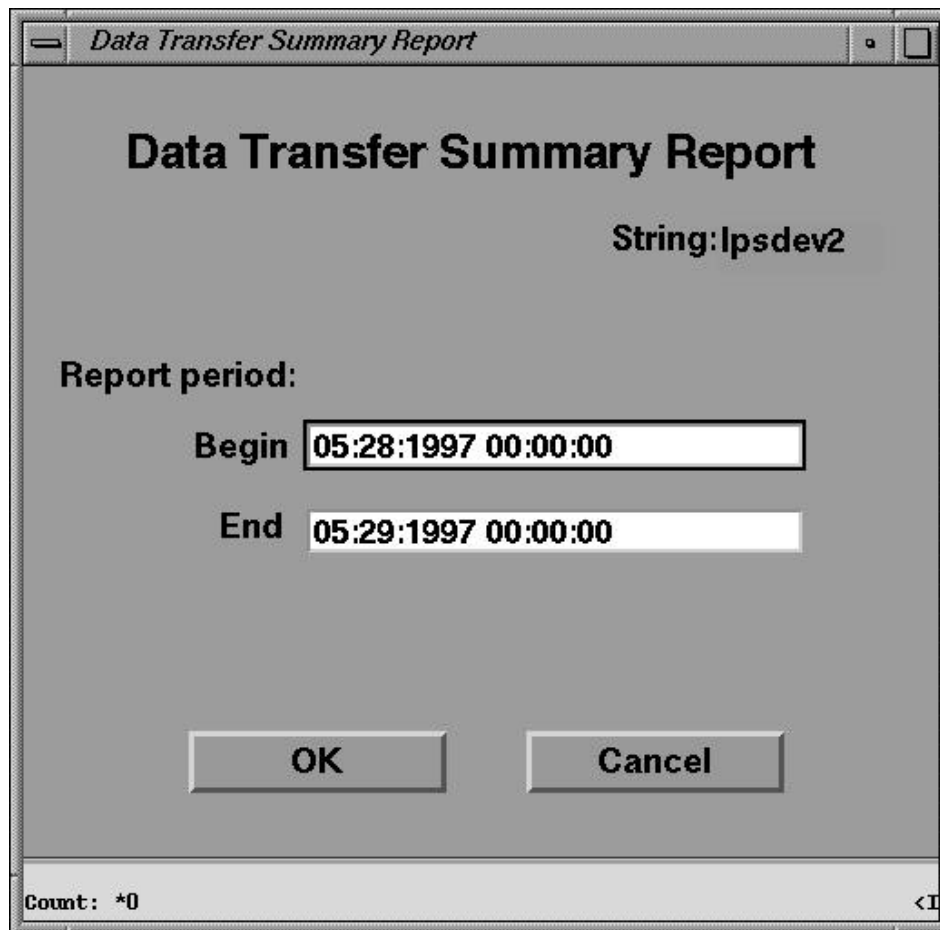*Figure 10–26. LPS GUI LPS Periodic Quality/Accounting Report Dialog*

Figure 10–27. LPS GUI Data Transfer Summary Report Dialog
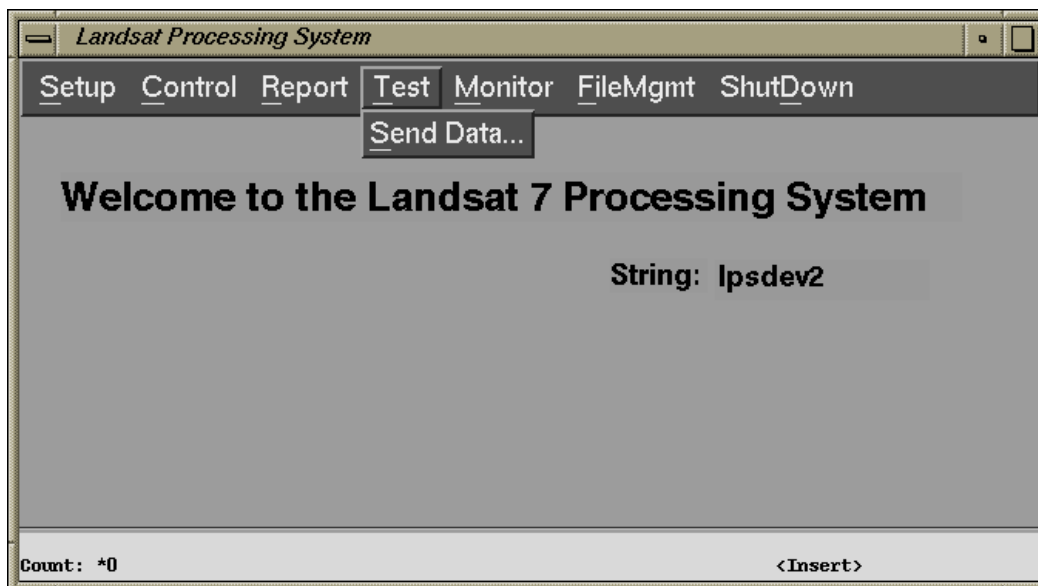

Figure 10–28. LPS GUI Test Menu

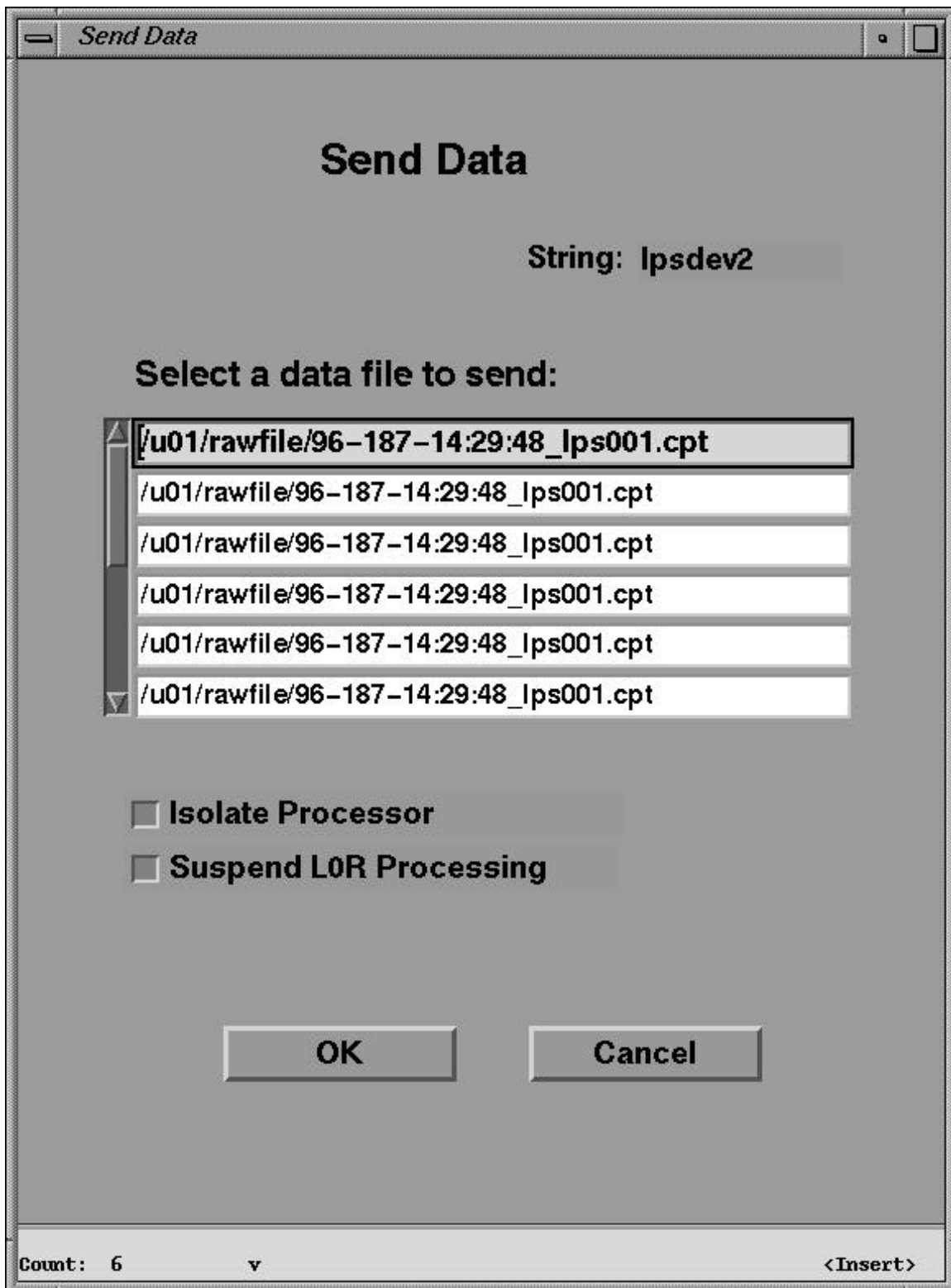*Figure 10–29. LPS GUI Send Data Dialog*
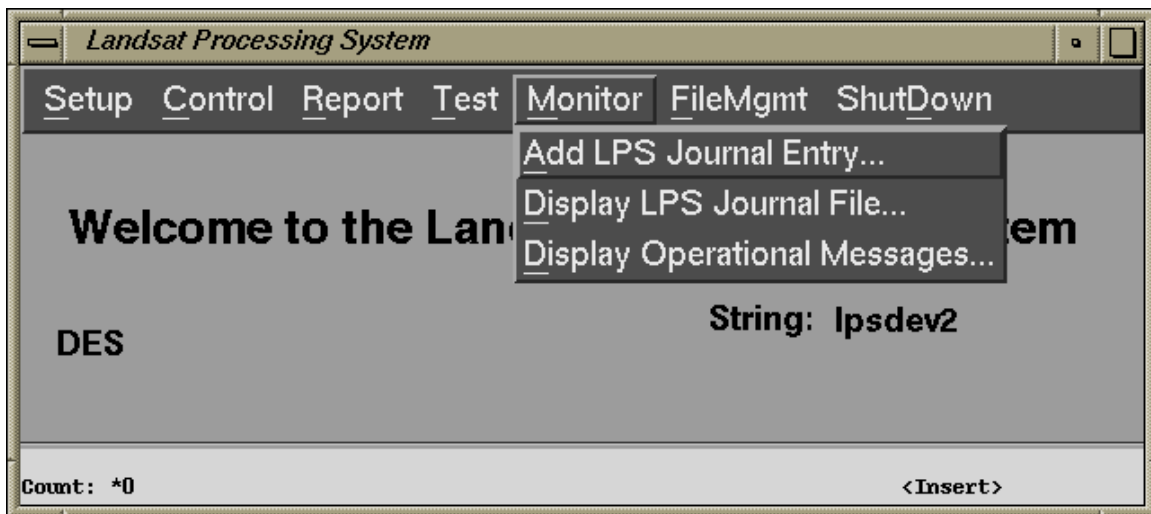
Figure 10–30. LPS GUI Monitor Menu
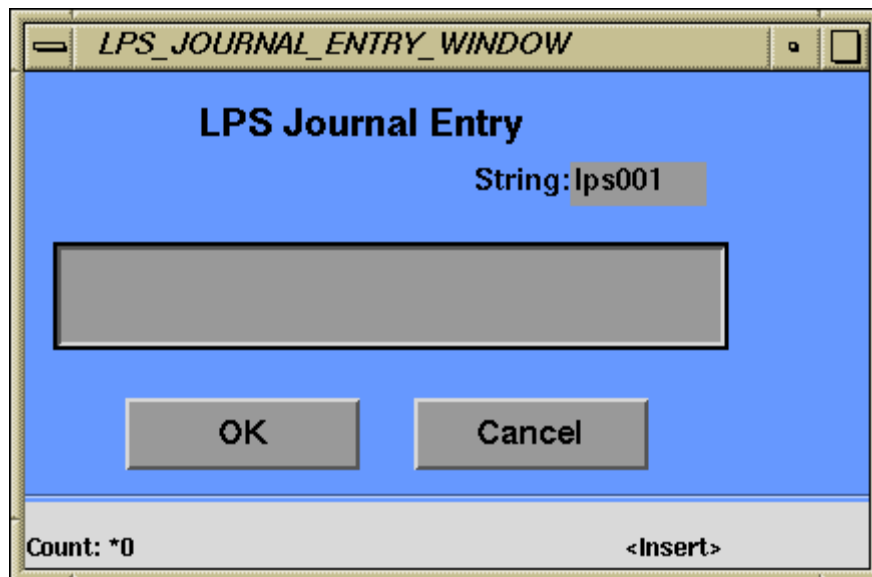
Figure 10–31. LPS GUI Journal Entry Window
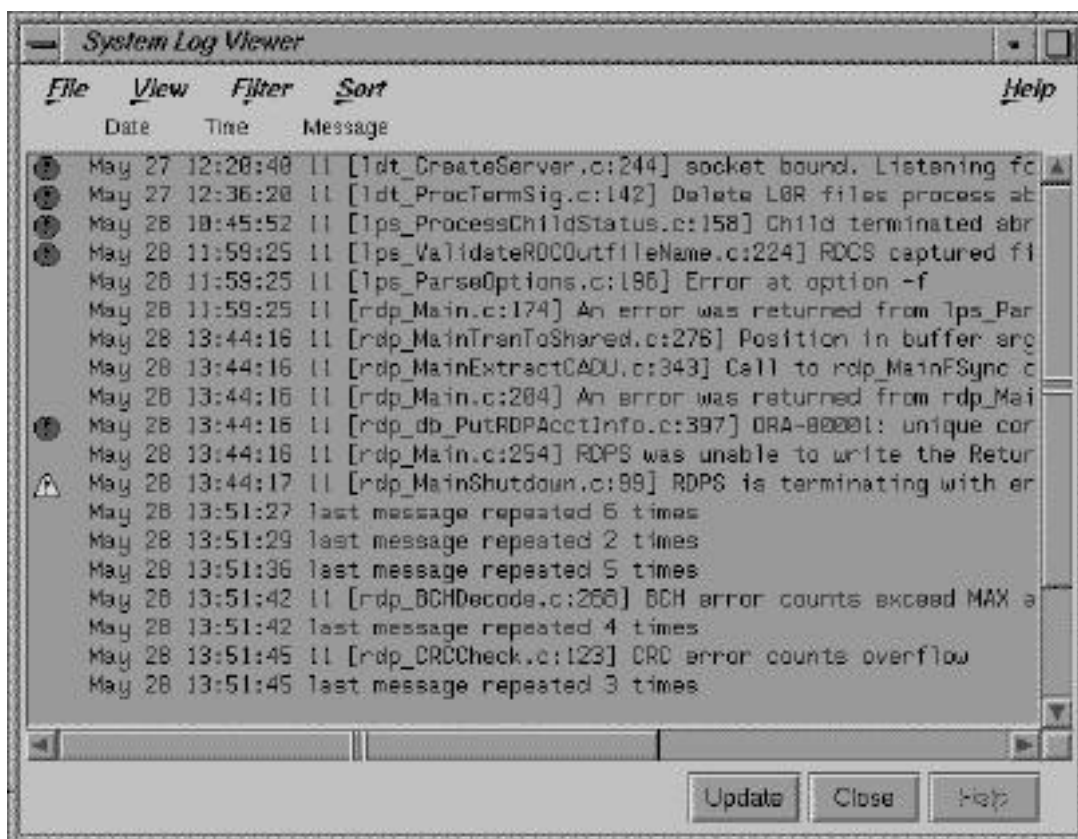
Figure 10–32. LPS GUI Journal File Display Window

Figure 10–33. LPS GUI Display Operational Messages Dialog
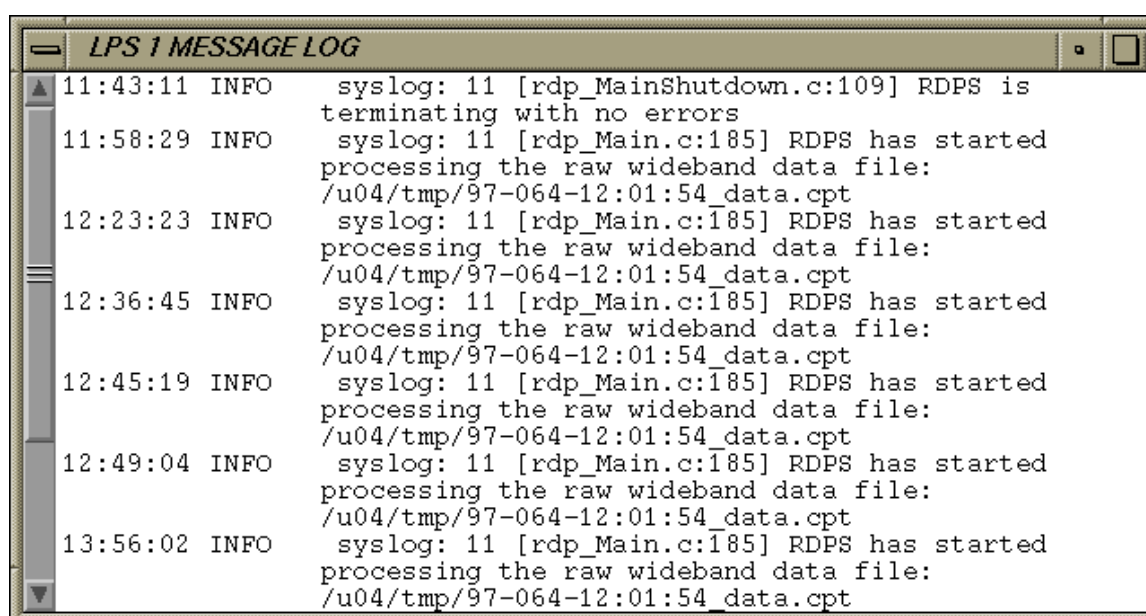
Figure 10–34. LPS GUI Operational Messages Display Window
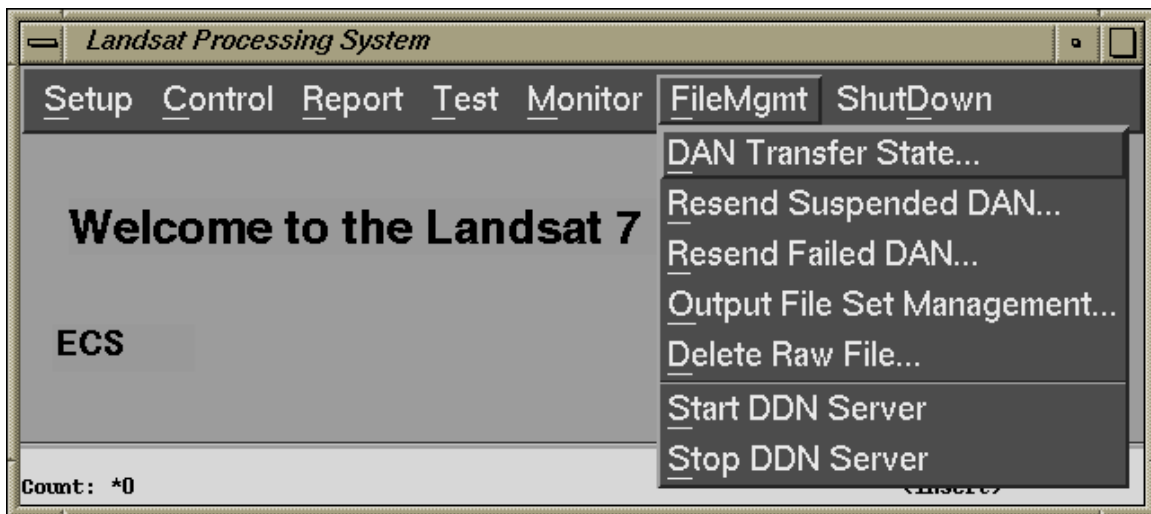
Figure 10–35. LPS GUI File Management Menu

Figure 10–36. LPS GUI Set DAN Transfer State Dialog

Figure 10–37. LPS GUI Resend Suspended DAN Confirmation Dialog
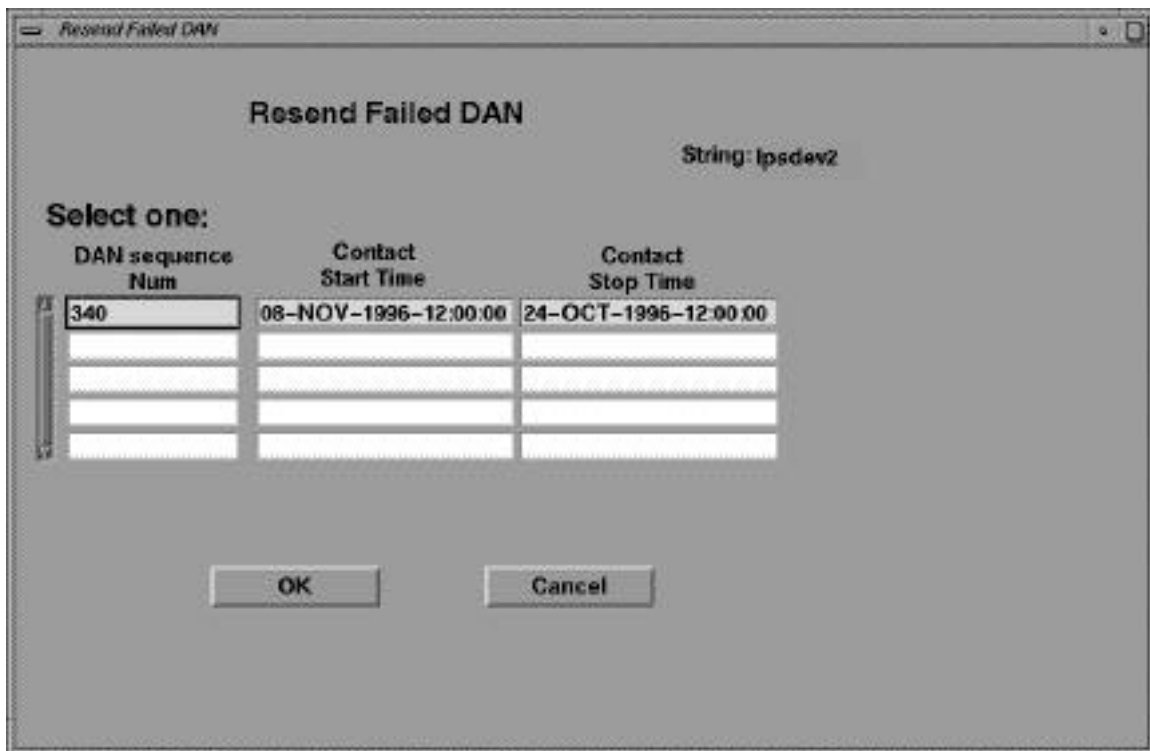
Figure 10–38. LPS GUI Resend Failed DAN Dialog
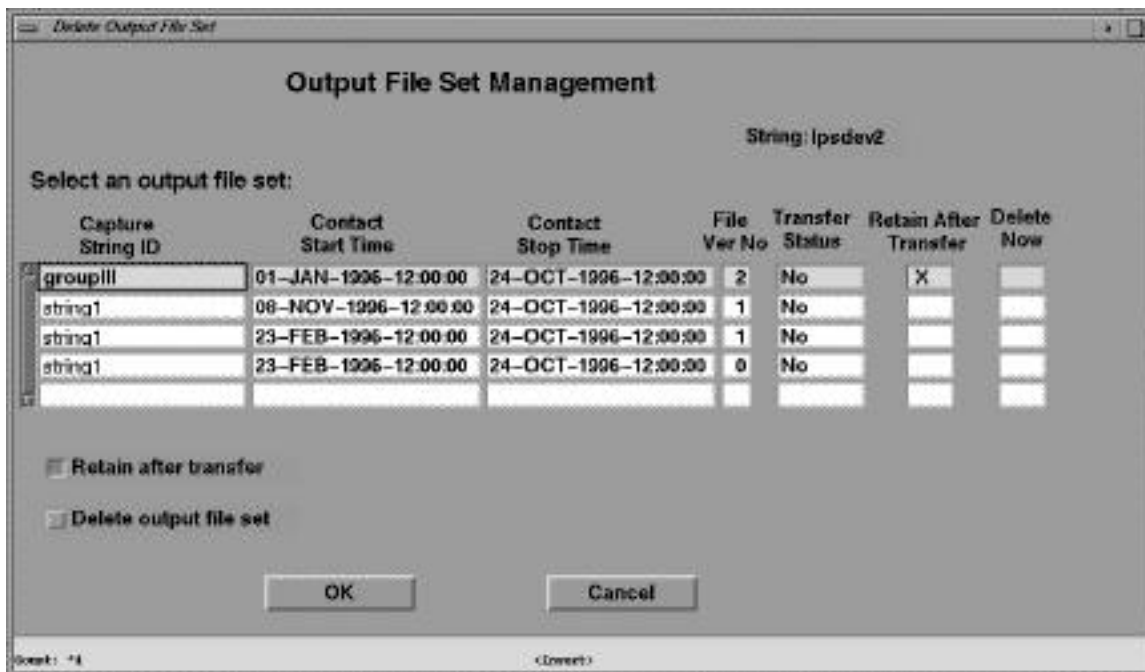
Figure 10–39. LPS GUI Output File Set Management Dialog
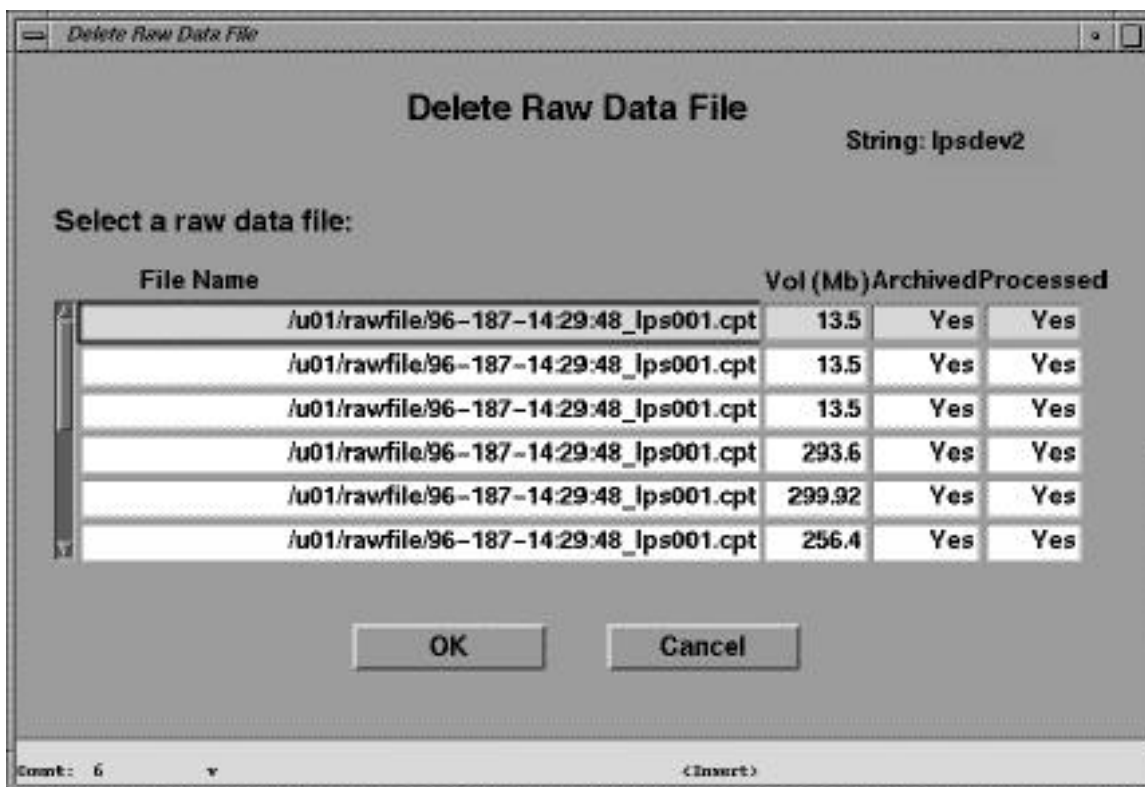
Figure 10–40. LPS GUI Delete Raw Data File Dialog

Figure 10–41. LPS GUI Stop DDN Server Confirmation Dialog
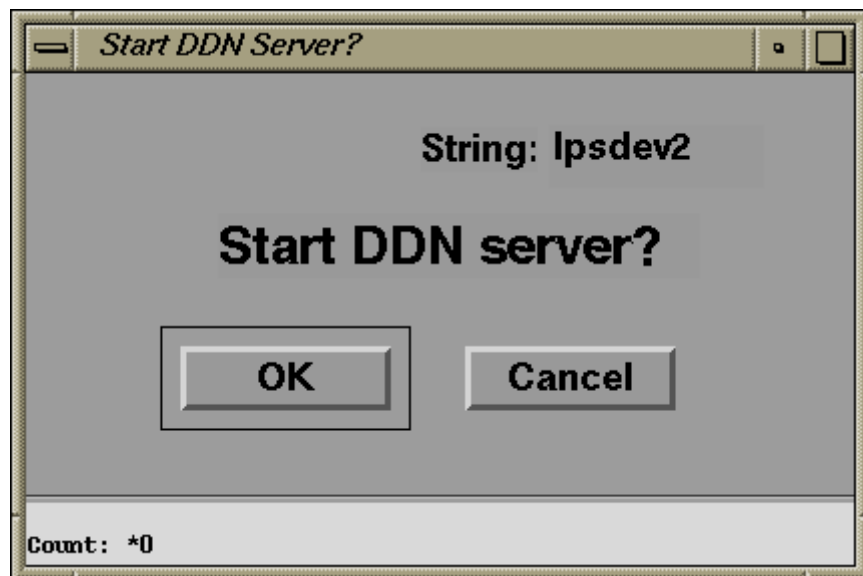
Figure 10–42. LPS GUI Start DDN Server Confirmation Dialog
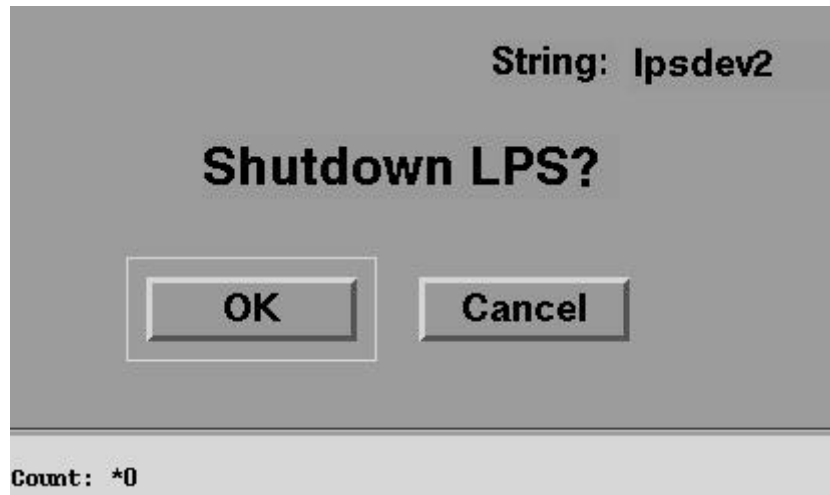
Figure 10–43. LPS GUI Shutdown LPS Confirmation Dialog
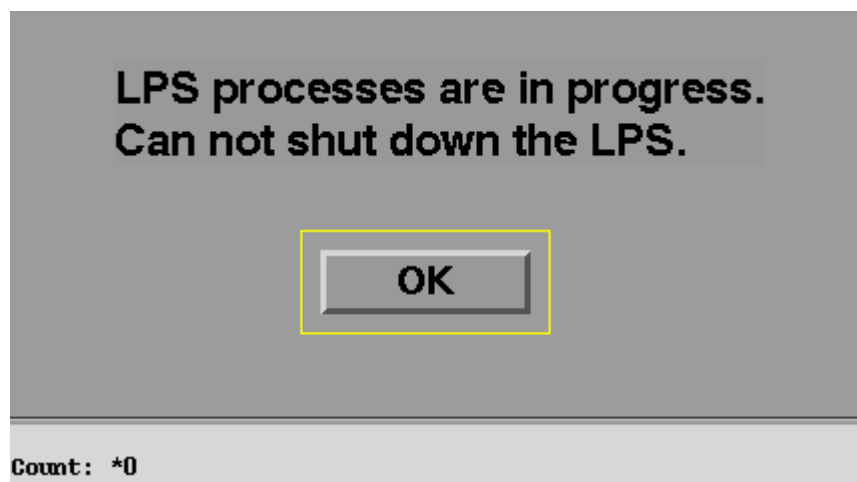
Figure 10–44. LPS GUI Shutdown Command Failure Dialog

Appendix A. LPS Error Messages

A.1 Introduction

This appendix contains a list of error messages generated by the LPS software. The text of each error message appears along with the message's severity code, a description of the error's cause, and, where appropriate, instructions for correcting the error. The line number of the error messages is not included in this appendix.

Oracle server messages can be found in *Oracle 7 Server Messages and Codes Manual* (Reference 6), Oracle forms messages can be found in *Oracle Developer/2000 Forms 4.5 Messages and Codes Manual* (Reference 7), and Oracle reports messages can be found in *Oracle Developer/2000 Reports 2.5 Messages and Codes Manual* (Reference 8).

A.2 Message Format

Each message has an entry with the following components:

- Priority: Message's IRIX priority, both the symbolic name and the numeric value

- Cause: Condition that causes the LPS software to generate the message

- Consequences: Effect the condition causing the message has on LPS software. The consequences are of most interest for error conditions. In this case, this entry states whether or not the LPS process continues to run and what actions have been taken to correct or compensate for the error (e.g., using a default value when an illegal one is supplied on the command line).

- Recommended action: What the user needs to do when the error is encountered

The messages are grouped by subsystem.

A.3 LPS Status and Error Messages

See Volume 2.

Appendix B. Directory Structure and File Name Formats

This appendix describes the structure of the directories on LPS strings containing files of interest for LPS processing and the formats for LPS file names.

B.1 String Directory Structure

The string directory structure is defined in *Landsat 7 Processing System (LPS) Software Configuration Guide* (Reference 9).

B.2 Output File Subdirectory Structure

LPS L0R processing software creates a new subdirectory in the directory pointed to by the LPS_OUTFILE_PATH environment variable for each instance of L0R processing. Within that directory, the software creates a separate directory for each subinterval extracted from the raw wideband data file. All output files generated during a particular L0R processing run are created in the directory for the subinterval to which they belong. The LPS_OUTFILE_PATH directory, therefore, has the structure shown in Figure B–1.

Contact Id is the contact ID assigned to the raw wideband data file being processed by the LPS database. The format for *Contact Id* is contact-XXXXX, where X ranges from 0 to 9. *File Version No.* is the file version number assigned to the L0R processing run. The format for *File Version No.* is ver-X, where X ranges from 0 to 9. *Subinterval Id* is the subinterval ID assigned to the subinterval by the LPS database. The format for *Subinterval Id* is subint-XX, where XX ranges from 01 to 99.

B.3 Raw Wideband Data File Name Format

Raw wideband data file names have the following format:

    YY-DDD-HH:MM:SS_String.data

YY is the last two digits of the year of capture. *DDD* is the Julian day of capture. *HH:MM:SS* is the time of capture (hour, minute, second) using a 24-hour clock. *String* is the host name (for example, lps001) of the LPS string that captured the data in the file.

B.4 L0R Output File Name Format

L0R output file names have the following format:

    L7XsssfnYYDOYHHuuv.xxs.

X is the Landsat 7 X band (1, 2, or 3) on which the data was transmitted. *sss* is the data capture ground station ("EDC" for the EROS Data Center). *f* is the ETM+ format (1 or 2). *n* is the LPS processor number (the last character of the host name; lps001 = processor number 1, for
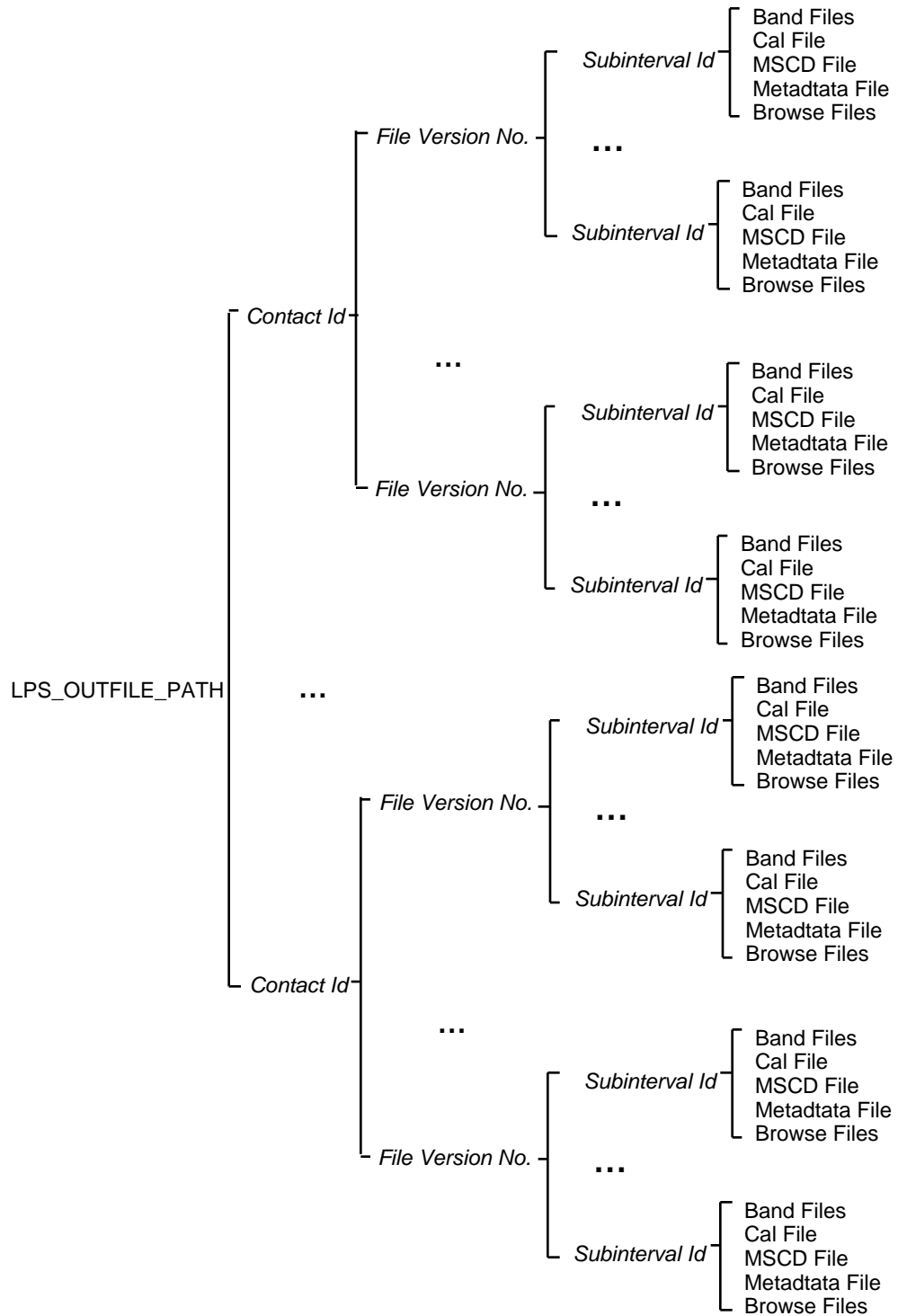
Figure B–1. LPS Output File Directory Hierarchy

example). *YY* is the last two digits of the year of capture. *DOY* is the Julian day of capture. *HH* is the hour of the capture time. *uu* is the sequence number of the subinterval among others in the contact (01 to 99). *v* is the file version number (0 to 9). *xxs* is the file type: B1, B2, B3, B4, B5, B6, B7, B8 for band files, MSD for MSCD files, PCD for PCD files, CAL for calibration files, MTA for metadata files, Rnn for browse files, nn = the sequence number of the scene within the subinterval (01 to 99).

B.5 Trouble File Name Format

L0R trouble file names have the following format:

> *RawFileName_System_*TroubleFileV*FileVersionNumber*

RawFileName is the name of the raw wideband data file being processed (see Section B.3 for the raw file name's format). *System* is the LPS software subsystem that generated the file. If "RDPS" is the value for *System*, the trouble file contains transfer frames with quality annotations. If "MFPS" is the value for *System*, the trouble file contains sets of transfer frames with annotations plus additional annotations related to the attempt to produce an ETM+ major frame from them. *FileVersionNumber* is the file version number assigned to the instance of L0R processing that produced the file.

B.6 LPS/EDC DAAC Message File Name Formats

LPS/EDC DAAC messages are saved in files with names in the following format:

> *Type_NNNNNNNNNN_MM*.msg

Type is the message type: DAN, DAA, DDN, or DDA. *NNNNNNNNNN* is the DAN sequence number associated with the set of messages. *MM* is the message number.

The formats of the names of authentication messages are as follows:

- For a request: *AUTHREQT_(pid)_MM.msg*

- For a response: *AUTHRESP_(pid)_MM.msg*

pid is the process ID for senddan (request) or rcvddn (response). *MM* is the message number.

Appendix C. Process Catalog and Man Pages

This appendix describes the function of each LPS software process. Descriptions are ordered by process name. The list includes not only LPS application software processes, but system software processes that have a significant role in performing LPS functions. The list does not include Forms 4.5 executables or interpreter scripts (such as those used by awk).

Each process is accompanied by an indication of whether or not it can be invoked as a standalone program from an IRIX shell. Further information on LPS standalone programs can be found in Appendix E. All the standalone programs have a man page. The man page may be evoked from the IRIX shell as follows:

```
% man deletefile
```

If desired, a copy of the man page may be piped into an ASCII file with the aman command as follows:

```
% aman deletefile > deletefile.asc
```

Process Name	Standalone?	Description
deletefile	Y	Deletes LPS L0R output product files; invoked by rcvddn or by LPS operator
genfts	N	Generates LPS File Transfer Summary Report
idp	N	L0R process that controls the generation of band files and browse files, performs ACCA, and generates the MWD
idp_acca	N	L0R process that performs ACCA
idp_band	N	L0R process that generates band files
idp_browse	N	L0R process that generates browse files
idp_mwd	N	L0R process that displays scan data in a Moving Window
mac_autocapture	Y	Automatically captures data, performs L0R processing, and archives raw data according to schedule
mac_db_init	Y	Updates online status of raw wideband data files in LPS database by checking raw data file storage array
mac_lps	Y	LPS startup process
mac_startl0r	Y	Controls L0R processing of a specified raw wideband data file
mac_ui_filtermsg.ser	Y	Filters and formats messages as they are written to the LPS Journal file

Process Name	Standalone?	Description
mac_ui_opsmsg.ser	Y	Generates a formatted display of LPS status and error messages as they are written to the LPS Journal file
mfp	N	L0R process that constructs ETM+ frames and generates calibration and MSCD files
pcd	N	L0R process that constructs PCD frames, generates PCD files, and identifies WRS scene centers
rcvddn	Y	DDN server – accepts and processes DDNs from EDC DAAC; deletes successfully transferred files, notifies operator of file transfer errors, and updates output file transfer accounting in the LPS database
rdc_Capture	Y	Captures raw wideband data beginning immediately and continuing for a specified period of time to disk
rdc_DeleteFiles	Y	Deletes a specified raw wideband data file and updates appropriate accounting information in the LPS database
rdc_GenLabel	Y	Generates a tape label for the specified raw wideband data file using appropriate accounting information in LPS database
rdc_Restage	Y	Extracts specified raw wideband data file and its corresponding accounting file from a tape
rdc_Save	Y	Saves specified raw wideband data file and its corresponding accounting file onto DLT library
rdc_Terminate	Y	Terminates all processes with specified name
rdc_Transmit	Y	Transmits raw wideband data
rdc_UpdRDCAcct	Y	Searches for capture accounting information files in reserved directory and updates database
rdc_vmereset	Y	Resets the Versa-Module European (VME) boards on a specified VME bus
rdp	N	L0R process that performs return-link processing to extract and validate transfer frames from raw wideband data
retainfile	N	Retain LPS L0R output file products
rsndfaileddan	N	Resend a failed DAN to EDC DAAC for a specified contact period
rsndsuspdan	Y	Resend all suspended DANs to EDC DAAC after LPS/ EDC DAAC file transfer operation resumes
senddan	Y	Send a DAN to EDC DAAC for specified L0R processing run

Process Name	Standalone?	Description
stopddn	Y	Stop DDN server
syslogd	N	IRIX system log daemon process – accepts LPS status and error messages, formats and time tags them, and routes them to the LPS Journal file
xwsh	Y	IRIX X Window shell program; invoked by monsym to produce a window to display LPS status and error messages

Appendix D. Customizing Your Environment

The user normally will not need to manipulate environment elements directly. The LPS .lpsrc file will automatically set up an environment (see Section 2.3). This appendix explains what environment variables are important for LPS operations and how to customize their values, should that be necessary. Specifically, this appendix covers the following topics:

- Setting up to use a different LPS software version

- Customizing environment variables within a particular version

- LPS-specific environment variables

- ORACLE environment variables

- Shell environment variables

NOTE

Your environments on each LPS string and operator interface workstation are independent. Any change must be made separately on each LPS string and workstation.

D.1 Setting Up to Use a Different LPS Software Version

To use a different version of LPS software, replace the definition of LPS_HOME at the top of your .cshrc file with the name of the top-level directory of the LPS software version you wish to use. Table D–1 lists the LPS software version currently installed on the LPS003 string.

Table D–1. LPS Software Version

Top-Level Path	Description
usr/LPS/at/r2	Release 2 acceptance test
usr/LPS/test/r#	where test might be "st" or "at" and r# is release number

D.2 Customizing Environment Variables Within a Particular Version

There are two basic approaches for customizing your environment:

1. Replace the standard .lpsrc file with your own custom version.

2. Override the standard environment variable values.

The first approach is effective when you will be changing the values of many environment. If you are making only a few changes, the second approach is probably more effective.

To replace the standard .lpsrc file with your own custom version, do the following on each LPS string and workstation:

1. Make a copy of the standard .lpsrc file in your home directory by typing:

    ```
    % cp $LPS_HOME/.lpsrc ~
    ```

2. Use a text editor to edit your copy of .lpsrc to change the values assigned to environment variables.

3. Use a text editor to edit your .cshrc file to change your sourcing of .lpsrc to the following:

    ```
    source ~/.lpsrc
    ```

To override standard environment variable settings, do the following on each LPS string and workstation:

1. Use a text editor to edit your .cshrc file.

2. After the "source $LPS_HOME/.lpsrc" line, add lines setting the environment variables you want to customize as follows:

    ```
    setenv Environment-Variable-Name Custom-Value
    ```

    For example, to override the standard settings of LPS_REPORT_PATH and LPS_TROUBLE_FILE_PATH to write reports and trouble files into two subdirectories of your home directory, you would add the following to your .cshrc file:

    ```
    setenv LPS_REPORT_PATH ~/myreports
    setenv LPS_TROUBLE_FILE_PATH ~/mytrouble
    ```

                                NOTE

>   Setting environment variables from the command line is not effective for commands and programs executed in subprocesses. The standard .lpsrc file is executed for every subprocess, resetting environment variables to their standard values. In particular, custom values will not be in effect when LPS software is activated unless the settings are made in your .cshrc file AFTER .lpsrc has executed.

D.3 LPS-Specific Environment Variables

The LPS software uses the environment variables listed in Table D–2. You can set these values to control the associated LPS behavior. Guidelines for setting values and restrictions on acceptable values appear in the descriptions.

D.4 ORACLE Environment Variables

The LPS software assumes that environment variables required by the ORACLE DBMS have been correctly defined. Table D–3 lists the ORACLE environment variables required for the LPS software.

For a detailed discussion of ORACLE environment settings consult *Oracle 7 for MIPS ABI Systems Installation and Configuration Guide* (Reference 10).

D.5 Shell Environment Variables

The LPS software makes use of the standard shell environment variables listed in Table D–4.

In addition, defining the environment variable $IDPS_DEBUG in the user's shell will cause the IDPS to dump the content of the HDF error stack to two files:

- /tmp/idp_band.hdf.errs

- /tmp/idp_browse.hdf.errs

This is useful for tracking down problems if errors result when writing the band or browse files.

Table D–2. LPS Environment Variables (1 of 3)

Environment Variable	Description
LPS_BIN	Directory where the LPS software searches for executable images. This should be set to the path name of the directory containing the version of the LPS software you want to run. Normally set to $LPS_HOME/bin.
LPS_DANFILE_PATH	Directory into which the LPS software writes a copy of each DAN it sends to EDC DAACs.
LPS_DDNFILE_PATH	Directory into which the LPS software write a copy of each DDN it receives from EDC DAAC.
LPS_HOME	Top-level directory containing the LPS software installation. To set up so that your environment points to a version of LPS software automatically, explicitly set this variable in your .cshrc file just prior to the "source $LPS_HOME/.lpsrc" line.
LPS_JOURNAL_PATH	Directory containing the LPS Journal file. WARNING: Changing the value of this variable does not change the directory in which the LPS Journal file resides. The location and name of the LPS Journal file is specified in the system file, /et/syslog.conf, as the destination for local0 facility messages. This variable should always be set to the same file.
LPS_LOG_STDOUT	When set to any value, indicates that the LPS software should write status and error messages to stdout, as well as to the LPS Journal file. This variable should be set only when LPS software is being executed from a UNIX shell. WARNING: Running LPS software through the LPS GUI when this variable is set could hang the LPS software.

Environment Variable	Description
LPS_OUTFILE_PATH	Top-level directory for LPS output files. Should be set to the top-level directory of the file system on the LPS string's output redundant arrays of independent disks (RAID) device. Your user ID must have read permission for the directory.
LPS_PRINTER_DEVICE	Specifies the printer device to which ("/dev/plp") to print the tape labels.
LPS_PROCESSOR_NUMBER	Processor to isolate and restrict a process to this processor. If LPS_PROCESSOR_NUMBER is invalid or NULL, the #RDC_DEFAULT_PROCESSOR number of 1 is used.
LPS_RAWFILE_PATH	Used to verify adequate disk space for raw data capture. Destination directory of the raw wideband data files and the associated accounting information files during data capture: $LPS_RAWFILE_PATH/*filename*.data (raw wideband data) $LPS_RAWFILE_PATH/*filename*.acct (accounting) Should be set to the top-level directory of the file system on the LPS string's input RAID device. Your user ID must have read permission for the directory. If not defined in the user's environment, "./" (current directory) is used as the default.
LPS_REPORT_PATH	Directory into which the LPS software will write files containing LPS reports. Your user ID must have read and write permission for the directory.
LPS_TABLE_PATH	Directory holding files containing static data loaded by the LPS software during its execution. Normally, set to $LPS_HOME/tables.
LPS_TAPE_DEV	The name of the default tape device for copying raw wideband data files to the 30-day store. The default is /dev/rmt/tps131d5.
LPS_TAPE_LIBRARY_DEV	Specifies the tape device (jukebox). The default is /dev/scsi/sc131d510.
LPS_TEMPFILE_PATH	Directory into which the LPS software will create temporary files. Used to verify adequate system disk space for raw data capture. It is assumed that this environment variable will point to a directory resident on the system disk. Your user ID must have read permission for the directory. If not defined in the user's environment, "./" (current directory) is used as the default.
LPS_TROUBLEFILE_PATH	Directory into which the LPS software will write trouble files. Your user ID must have read and write permission for the directory.

Table D–2. LPS Environment Variables (3 of 3)

Environment Variable	Description
RDC_DEVICE	Specifies the input/output device name used to receive raw wideband data. If not defined in the user's environment, the value "/dev/hpdiB" is used as the default.
RDC_STATUS_INTERVAL	Specifies the reporting interval during raw wideband data capture (seconds). If not defined in the user's environment, the value "30" is used as the default.
RDC_THRESH_SYSTEMDISK	Specifies amount of available system disk space that must exist before reporting warnings (MB). This environment variable is used when checking disk space. If not defined in the user's environment, the value "0.01" is used as the default.

Table D–3. ORACLE Environment Variables for LPS Software

Environment Variable	Description
FORMS45_PATH	Path name of directory containing LPS GUI Forms executables; normally set to $LPS_HOME/bin
REPORTS25_PATH	Path name of directory containing LPS GUI Reports executables; normally set to $LPS_HOME/bin
ORACLE_HOME	Path name of top-level directory of the ORACLE installation containing the bin, lib, guicommon, formsXX, and reportsXX subdirectories
ORACLE_PATH	List of directories containing executables; similar to the UNIX PATH variable
ORACLE_SID	Name of database instance to which LPS software will connect
ORACLE_TERM	Terminal type
TK2_TERMINAL	Terminal characteristics file

Table D–4. Shell Environment Variables (1 of 2)

Environment Variable	Description
PATH	List of directories in which the shell will search for executables [see csh(1) for details of its operation]. PATH should include both $LPS_HOME/bin and $ORACLE_HOME/bin.
MAN_PATH	List of directories containing man pages [see man(1) for details of its operation]. MAN_PATH should include $LPS_HOME/man for LPS man pages and appropriate directories in $ORACLE_HOME for ORACLE man pages.

Table D–4. Shell Environment Variables (2 of 2)

Environment Variable	Description
DISPLAY	Address of X Window System display. DISPLAY should be set to "*Address*:0" where *Address* is the IP name or IP address of the X display device (X terminal or workstation).
LD_LIBRARY_PATH	Lists additional directories containing dynamic shared objects. The list should include the default directories (/lib & /usr/lib), as well as directories for ORACLE shared objects (consult Reference 10).

Appendix E. Performing LPS Functions Through IRIX

The LPS software is designed so that all major operations can be invoked from both the LPS GUI and any IRIX shell. Also, access to the LPS database is possible through both the LPS GUI and Oracle's SQL*Plus. This combination of capabilities allows the user to invoke LPS functions in ways not supported by the LPS GUI. In particular, it allows the user to create shell scripts that invoke LPS functions, thereby extending LPS capabilities.

NOTE

> Performing the functions possible only through IRIX or SQL*Plus, the instructions appear in the document's body. In these cases, this appendix contains only a reference to the section in the document.

E.1 Finding LPS Database Identifiers

The LPS software uses unique IDs to reference information about captured data and the output of L0R processing. To invoke LPS programs from IRIX, it is frequently necessary to supply these IDs. The following IDs are frequently needed:

- *Contact ID* – a number assigned automatically by the ORACLE DBMS to identify the data captured on a LPS string during a certain contact

- *File Version Number* – a number assigned by the operator to a particular L0R processing run

- *Subinterval ID* – a number assigned automatically by the ORACLE DBMS to identify a particular subinterval extracted from a raw wideband data file during a particular L0R processing run

For the IDs automatically assigned by the ORACLE DBMS, query the LPS database to determine what they are. For the file version number, you will frequently already know its value. However, it is possible to retrieve a list of file version numbers assigned to the processing runs on the string.

E.1.1 Finding a Contact Identifier

To find a contact ID, query the LPS database on the string storing the raw data capture file. You must know one of the following for the contact period:

- Scheduled capture start time

- Scheduled capture stop time

- Actual capture start time

- Actual capture stop time

- Raw data capture file name

Each LPS string assigns contact IDs to contacts independently. Raw wideband data files captured during the same contact by different strings may not have the same contact ID; files captured during different contacts may have the same IDs.

To retrieve the contact ID, type the following to SQL*Plus:

```
SQL> SELECT contact_sequence_id FROM rdc_acct WHERE condition;
```

The value you enter for *condition* will depend on what you know about the contact period. Table E–1 lists the values for *condition* for each of the items above. It also describes how to query for a list of contacts from which you can select the contact sequence ID.

Table E–1. Query Conditions for Selecting Contact Sequence IDs

If you know...	Then use this as the where clause
Scheduled capture start time	WHERE scheduled_start_time = TO_DATE('*Date&Time*', *'Format'*) *Format* must describe the format of Date&Time. For example, 'YY-DDD-H24:MI:SS.' See *Oracle 7 Server SQL Reference* (Reference 11) for details on format options for the to_date() function.
Scheduled capture stop time	WHERE scheduled_stop_time = TO_DATE('*Date&Time*', *'Format'*)
Actual capture start time	WHERE actual_start_time = TO_DATE('*Date&Time*', *'Format'*)
Actual capture stop time	WHERE actual_stop_time = TO_DATE('*Date&Time*', *'Format'*)
Raw data capture file name	WHERE raw_data_file_name = '*filename*'
Any field and want a list of all **unprocessed** contacts on the string's raw data capture storage	WHERE on_line_flag = 1 AND contact_sequence_id not in (SELECT UNIQUE contact_sequence_id FROM processing_version_info) Add any combination of scheduled_start_time, scheduled_stop_time, actual_start_time, actual_stop_time, and raw_data_file_name to the list of selected attributes so that the contact can be identified. For example, "select contact_sequence_id, raw_data_file_name from..."
Any field and want a list of **all** contacts on the string's raw data capture storage	WHERE on_line_flag = 1 Add any combination of scheduled_start_time, scheduled_stop_time, actual_start_time, actual_stop_time, and raw_data_file_name to the list of selected attributes so that the contact can be identified. For example, "select contact_sequence_id, raw_data_file_name from..."

E.1.2 Finding a File Version Number

Normally you will already know the file version number you want to use for a LPS operation. You can find a list of the file version numbers of L0R processing instances on this LPS string for a given contact by typing the following to SQL*Plus:

```
SQL> SELECT file_version_number FROM processing_version_info
```

```
        2>    WHERE contact_sequence_id = Contact Identifier;
```
Contact Identifier is the contact ID for the contact.

Example: Retrieve a list of file version numbers for the contact with contact ID = 86214.

```
        SQL> SELECT file_version_number FROM processing_version_info
        2>    WHERE contact_sequence_id = 86214;
        FILE_VERSION_NUMBER
        ------------------
                         0
                         1
                         5
```

On this string, the contact ID with ID 86214 has L0R processing instances with file version numbers of 0, 1, and 5.

You can produce a list of all file version numbers together with other information describing the contact by typing the following to SQL*Plus:

```
        SQL> SELECT file_version_number, scheduled_start_time,
        scheduled_stop_time,
        2>    actual_start_time, actual_stop_time, raw_data_file_name
        3>    FROM processing_version_info, rdc_acct
        4>    WHERE processing_version_info.contact_sequence_id =
        rdc_acct.contact_sequence_id;
```

E.1.3 Finding a Subinterval Identifier

To find a subinterval ID you must know L0R processing instance that produced the subinterval, as well as either the subinterval's start or stop time or the subinterval's ordinal number within the L0R processing instance.

1. Find the contact ID and the file version number of the L0R processing instance that contains the subinterval. Sections E.1.1 and E.1.2 provide instructions on how to do so.

2. Look up the database attribute name of a known value identifying the subinterval in Table E–2.

3. Type the following to SQL*Plus:

```
        SQL> SELECT sub_intv_sequence_id FROM sub_intv
        2>    WHERE contact_sequence_id = Contact Identifier
        3>    AND file_version_number = File Version Number
        4>    AND Attribute = Value;
```

Table E–2. Attribute Names for Subinterval Identifying Fields

Known Value	Attribute Name
Subinterval number within contact (1..37)	sub_intv_number
Subinterval start time Format = YYYY:DDD:HH:MM:SS.ttttttt YYYY = year DDD = Julian day of year HH = hours of a day (24-hour clock) MM = minutes of an hour SS = seconds of a minute ttttttt = decimal fraction of a second	mf_start_time
Subinterval stop time Same format as start time	mf_stop_time

Contact Identifier and *File Version Number* are the contact ID and file version number, respectively, you found in step 1. *Attribute* is the attribute name you found in step 2. *Value* is the value you know. Note that if you are using either subinterval time, you must type the time in the format described in Table E–2 and enclose the time in single quotes.

1. Using one of the methods described in Sections E.1.1 and E.1.2, the contact ID is discovered to be 86214 and the file version number is 0.

2. In Table E–2, the attribute containing the subinterval start time has the name "mf_start_time."

3. Type the following to SQL*Plus:

```
SQL> SELECT sub_intv_sequence_id FROM sub_intv
2>    WHERE contact_sequence_id = 86214
3>    AND file_version_number = 0
4>    AND mf_start_time = '1998:210:16:24:34.2940111';
SUB_INTV_SEQUENCE_ID
--------------------
             86325
1 row(s) selected.
```

The subinterval ID is 86325.

E.2 Status and Error Message Monitoring

This section describes how to monitor LPS status and error messages from IRIX. All LPS status and error messages are written to the LPS Journal file.

- • Section E.2.1 describes the format of messages in the LPS Journal file.

- • Section E.2.2 describes how to set up windows that display messages as they are written to the LPS Journal file. It also describes how to construct custom filters that will select messages with specific priorities and format the messages for more convenient display.

- • Section E.2.3 describes how to set up your environment so that LPS status and error messages will be written in the window from which a LPS program was invoked.

- • Section E.2.4 describes how to review the contents of the LPS Journal file using IRIX utilities such as pagers and editors.

E.2.1 Message Format

The LPS software uses the IRIX system log daemon, syslogd, to route status and error messages to the LPS Journal file. Most of the formatting for LPS messages is performed by syslogd. See syslogd(1M) in Reference 2 for details.

The following is an example of a line from the LPS Journal file:

```
Apr 2 15:36:22 1Q:lps001 syslog: 3475 [rdc_Main.c:208]
Suspending level 0R processes
```

The date and time of the message appear first, followed by the priority, facility, and hostname (for "1Q:lps001" in the example, '1' is the priority, 'Q' is the facility, and 'lps001' is the host). The source is the next item. For messages generated by the LPS software, the source is always "syslog." The process group ID of the process generating the message appears next ("3475" in the example). The unit name and source line appear next in square brackets. These are normally of interest only for software troubleshooting. The final part of the line is the message itself.

The priority is represented as a number between 0 and 7 using the standard IRIX priority scheme. Table E–3 summarizes the priorities and their meanings. Note that smaller numbers indicate higher priorities.

IRIX facility codes are represented as "A" through "T." The facility code for all LPS messages is "Q." This corresponds to the local0 facility [see logger(1) in Reference 2].

Normally, you will create a status/error message display window for each LPS string by selecting menu options from the LPS GUI. It is also possible to view LPS messages as they are written to the LPS Journal file from an IRIX shell.

Table E–3. IRIX Message Priorities

Priority No.	Description
0	EMERGENCY – system is unusable
1	ALERT – immediate action must be taken
2	CRITICAL – critical condition
3	ERROR – error condition
4	WARNING – warning condition
5	NOTICE – normal but significant
6	INFO – informational message
7	DEBUG – debug level messages intended for software troubleshooting; not of operational interest

E.2.2 Setting Up Message Displays

To view LPS software status and error messages in real time from a UNIX shell, type the following commands:

```
% set LPSJournal = \
? `grep local0.debug /etc/syslog.conf | awk '{print $NF}'`
% cat –f $LPSJournal
```

To view LPS software status and error messages in real time in a separate X window, type the following commands:

```
% set LPSJournal = \
? `grep local0.debug /etc/syslog.conf | awk '{print $NF}'`
% xwsh –autofork –hold –e cat –f $LPSJournal
```

See (1G) in Reference 2 for details on ways to customize the window.

By using standard IRIX filters, it is possible to customize your display. You can filter out low-priority messages and display only the fields of interest. See Section 2.7.1 for details of LPS message format.

Figure E–1 is an example of a custom filter. The awk program filters out DEBUG (7) and INFO (6) priority messages and displays only the time, priority (converted to a four-character mnemonic), and the message itself, excluding the unit name, line number, and process group ID.

```
BEGIN {

      priorities[0] = "EMRG"

      priorities[1] = "ALRT"

      priorities[2] = "CRIT"

      priorities[3] = "ERRO"

      priorities[4] = "WARN"

      priorities[5] = "NOTE"

}

substr($4,1,1) < 6 {

      printf "%s %s ", $3, priorities[substr($4,1,1)];

      for (i = 8; i <= NF; ++i)

        printf "%s ", $i;

      printf "\n";

}
```

Figure E–1. Sample awk Script to Filter LPS Status and Display Messages

If this program was stored in the file awk.script, typing the following would produce a custom-formatted real-time display in a separate window:

```
% set LPSJournal = \

? `grep local0.debug /etc/syslog.conf | awk '{print $NF}'`

% xwsh –autofork –hold \

? -e "tail -f $LPSJournal | awk -f awk.script"
```

Note the double quotes surrounding the argument to the -e option. They are necessary to force the shell to interpret "| awk -f awk.script" as part of the option rather than as directing that the output of xwsh be piped to awk.

E.2.3 Directing Status and Error Message Display to the Standard Output

When running LPS programs directly from a shell rather than through the LPS GUI, you can direct the LPS software to write all status and error messages to the standard output (stdout) as well as to the LPS Journal file. This allows you to view the status and error messages from a process in the window from which it was invoked. The LPS program's output can be piped to other programs to filter and format the output. It can also be redirected to a file using csh(1) input/output redirection commands.

To direct LPS messages to the standard output, set the value of LPS_LOG_STDOUT to 1.

```
                    ┌─────────────────┐
                    │    WARNING      │
                    └─────────────────┘
```

> Processes invoked through the LPS GUI may hang when
> LPS_LOG_STDOUT is set. LPS_LOG_STDOUT should *never* be
> set in the shell from which the LPS GUI is invoked.

E.2.4 Browsing the LPS Journal

IRIX includes the sysmon(1M) utility for viewing system log files that can be used to browse the LPS Journal file. To browse the LPS Journal file using sysmon(1M), type the following:

```
% set LPSJournal = \
? `grep local0.debug /etc/syslog.conf | awk '{print $NF}'`
% sysmon -f $LPSJournal &
```

Information on the sysmon(1) utility is available in sysmon(1M) in Reference 2. The utility also provides online help.

The LPS Journal file is a text file and therefore can be browsed using any of the standard IRIX file display utilities, such as cat(1), more(1), and page(1). For example, to browse the LPS Journal file using more(1), type the following:

```
% set LPSJournal = \
? `grep local0.debug /etc/syslog.conf | awk '{print $NF}'`
% more $LPSJournal
```

The LPS Journal file can also be viewed using text editors. The file should not be locked or modified while the LPS software is running. To enter notations into the LPS Journal file while the LPS software is active [for example, to mark a change of shift, use the IRIX logger(1) command]. This avoids possible message corruption due to multiple, concurrent writers and ensures that message formats are correctly preserved.

E.3 Data Capture Operations

This section describes how to capture raw wideband data by invoking LPS capture software from an IRIX shell. This capability is intended for use during testing and for emergency situations in which data must be captured, but the LPS GUI and daemon processes are not functional (for example, when the ORACLE database server has failed).

- Section E.3.1 describes how to start data capture from IRIX.

- Section E.3.2 describes how to stop data capture from IRIX.

- Section E.3.3 describes how to restore LPS database consistency after capturing data when the ORACLE server is not running.

- Section E.3.4 describes how to set the capture source attribute in the LPS database directly through SQL*Plus.

• Section E.3.5 describes how to view and edit the contact schedule in the LPS database directly through SQL*Plus.

E.3.1 Starting Data Capture from an IRIX Shell

To capture data manually from an IRIX shell:

1. Create a new shell window or select an existing one on the LPS string that you want to capture data and move the cursor into the window's text area.

2. Type the following:

   ```
   % setenv LPS_LOG_STDOUT 1
   ```

   to direct status and error messages to the standard output (see Section 2.7.1.3). Messages will appear in this window.

   NOTE

   If a real-time message display is active (either created from the LPS GUI or a custom display you set up yourself as described in Section 2.7.1), capture status and error messages will appear there.

3. Invoke the LPS capture program, rdc_capture, by typing the following command:

   ```
   % rdc_Capture -l ID [-b Start-Time][-e End_Time][-s][-i] -o
   StationofOrigin
   ```

   The `rdc_Capture -l ID` option is optimal, but not required to capture data.

   <div style="border:1px solid black; display:inline-block; padding:4px 12px;">**WARNING**</div>

   The -l option should be used. If the correct value for the LGS channel is unknown, supply a valid dummy value (any number between 1 and 5) and correct it later in the LPS database.

   The remaining -b and -e options should be supplied if you know the correct values. The -b option indicates the scheduled start time for the contact being captured. The format should be "YYDDDHHMM:SS." The -e option indicates end time, in the same format as start time. The -s option causes rdc to attempt to suspend L0R processing before capturing data. The -i option causes rdc to attempt to run on a restricted, isolated, and nonpreemptive processor with highest nondegrading priority. The -o flag supplies the station of origin (e.g., EDC, SGS, or AGS).

   For more details on rdc_Capture and its parameters, consult the man page for rdc_Capture, available online.

4. Monitor messages from rdc to verify that capture is active.

E.3.2 Stopping Data Capture From an IRIX Shell

To stop data capture from an IRIX shell:

1. Create a new shell window or select an existing one on the LPS string that you want to capture data and move the cursor into the window's text area.

2. Type the following:

   ```
   % rdc_Terminate rdc_Capture
   ```

3. Monitor messages from rdc to verify that capture has terminated. You should see a message that resembles

   ```
   Apr 2 15:36:22 1Q:lps001 syslog: 3475 [rdc_Capture.c:208]
   Shutting down raw data capture
   ```

   If you invoked capture from an IRIX shell and set the LPS_LOG_STDOUT environment variable, the message will appear in the window in which you executed rdc. In any case, the message will appear in the LPS Journal file. For instructions on monitoring the LPS Journal file, see Section 2.7.1.

E.3.3 Cleaning Up After Capturing From an IRIX Shell

For Release 2, rdc_UpdRDCAcct can be used to update the database with the accounting information when the database is available.

E.3.4 Setting the Capture Source From ORACLE SQL*Plus

To determine the value of a string's capture source, type the following to SQL*Plus:

```
SQL> SELECT capture_source FROM lps_configuration;
```

The value returned is a two-character encoding of the possible sources of data. See Table 10–1 for explanations of their meanings.

To set the value of a string's capture source, type the following to SQL*Plus:

```
SQL> UPDATE lps_configuration SET capture_source =
'Value' ;
```

Value is a two-character encoding of the possible sources of data. See Table 10–1 for explanations of their meanings.

E.3.5 Updating the Contact Schedule From ORACLE SQL*Plus

This section describes how to ingest the contact schedule from the IRIX shell.

Invoke the LPS ingest procedure, mac_IngestContact, by typing the following command:

```
% mac_IngestContact file_name string_selection_flag
```

where *file_name* is the contact schedule file to be ingested and *string_selection_flag* is the 6-bit indicator of which string the file should be ingested. Each bit represents the following:

- The first bit is for LPS1/LPS (operational instance).

- The second bit is for LPS2/LPS (operational instance).

- The third bit is for LPS3/LPS (operational instance).

- The fourth bit is for LPS4/LPS (operational instance).

- The fifth bit is for LPS5/LPS (operational instance).

- The sixth bit is for LPS5/DEV (development instance).

For example, string_selection_flag = 101100 means to ingest into the first, third, and fourth string instances.

The contact schedule table contents can be modified directly using the SQL*Plus data manipulation language at the SQL*Plus prompt.

E.4 Raw Wideband Data Management

This section decsribes how to perform manage raw wideband data through by invoking LPS raw wideband data management software through an IRIX shell.

- Section E.4.1 describes how to copy raw data to tape from IRIX.

- Section E.4.2 describes how to stop copying raw data to tape from IRIX.

- Section E.4.3 describes how to restore raw data from tape to disk (restage) from IRIX.

- Section E.4.4 describes how to stop restage from IRIX.

- Section E.4.5 describes how to generate a tape label manually from IRIX.

- Section E.4.6 describes how to delete raw data files from IRIX.

E.4.1 Copy Raw Data to Tape

To copy data from to tape using the IRIX shell, invoke the LPS save program (rdc_Save) by typing the following command:

```
% rdc_Save [-b bin] filename
```

where the optional -b option is to provide the tape library bin number to perform archive and *filename* is the raw data filename to save to the archival device.

E.4.2 Stop Copying Raw Data to Tape

To stop the process of copying raw data to tape from an IRIX shell:

1. Create a new shell window or select an existing one on the LPS string from which you are copying data and move the cursur into the window's text area.

2. Type the following:

```
% rdc_Terminate rdc_Save
```

E.4.3 Restore Raw Data From Tape to Disk (Restage)

To restore raw data from tape to disk using the IRIX shell, invoke the LPS restage program (rdc_Restage) by typing the following command:

```
% rdc_Restage [-d device_name]
```

where the optional -d option provides the tape device from which to restage the data. (Slot 6 is the default used for the restage process.)

E.4.4 Stop Restoring Data From Tape to Disk

To stop the process of restoring raw data from tape to disk using the IRIX shell:

1. Create a new shell window or select an existing one on the LPS string from which you are copying data and move the cursur into the window's text area.

2. Type the following:

```
% rdc_Terminate rdc_Restage
```

E.4.5 Generate a Tape Label Manually

To generate a tape label manually from the IRIX shell, invoke the LPS restage program (rdc_GenLabel) by typing the following command:

```
% rdc_GenLabel -f /path/filename [-b slot_number]
```

where the -f option is needed to provide the raw wideband data filename and the optional -b option provides tape slot number.

E.4.6 Delete Raw Data Files

To delete raw data files from the IRIX shell, invoke the LPS rdc delete files program (rdc_Deletefiles) by typing the following command:

```
% rdc_DeleteFiles -f /path/filename [-u]
```

where -f /path/filename is the complete path and filename of file to delete. The -u option indicates unconditional, meaning both the accounting and the data file will be deleted. If -u is not specified (default), a conditional delete is performed. A conditional delete is when both the file has been L0R processed and saved to tape before it is deleted.

E.5 L0R Processing Management

This section describes how to invoke L0R processing from an IRIX shell.

- Section E.5.1 describes how to start L0R processing from IRIX.

- Section E.5.2 describes how to stop L0R processing from IRIX.

- Section E.5.3 describes L0R process parameters management for SQL*Plus.

- Section E.5.4 describes L0R error reporting management for SQL*Plus.

E.5.1 Starting L0R Processing

To perform L0R processing manually from an IRIX shell:

1. Identify the contact ID of the contact period you want to process. See Section E.1.1 for instructions.

2. Identify the file version number for the processing run you are going to start. For contacts that have never been processed, the file version number should be 0. The file version number should increment by 1 each time the contact is reprocessed. If you do not know the file version numbers that have been assigned to a contact, Section E.1.2 provides instructions to retrieve them from the LPS database.

> **WARNING**
>
> The LPS software will fail with an error if the file version number you supply has already been used on this LPS string. It will process the contact with the file version number you supply even if you skip a file version number or if the file version number has already been used on another LPS string. Consult *EDC Landsat 7 DHF Standard Operating Procedures* (Reference 12) for instructions for determining the file version number.

3. Verify that the LPS instance of the ORACLE DBMS server process is active by typing:

```
% ps -ef | grep ora | grep -v grep
```

If the ORACLE DBMS server process instance is active, the output from the command will resemble the following:

```
oracle 24623   1 0   May 20 ?     4:57 ora_dbwr_LPS
oracle 24626   1 0   May 20 ?     0:03 ora_reco_LPS
oracle 24624   1 0   May 20 ?     3:48 ora_lgwr_LPS
oracle 24622   1 0   May 20 ?     3:21 ora_pmon_LPS
oracle 24625   1 0   May 20 ?     0:41 ora_smon_LPS
```

If the command does not produce a list of process names, the ORACLE server instance for the LPS is not running and the LPS software cannot perform L0R processing. Contact the LPS system administrator and do not proceed until the ORACLE server instance for the LPS is active.

4. Start L0R processing by typing the following:

```
% mac_startl0r CSID Version [0 | 1] [0 | 1]
```

Use "1" as the second-to-last parameter if you want the LPS software to delete the raw data capture file when L0R processing completes successfully. Use "0" as the second-to-last parameter if you want to retain the file online after L0R processing completes successfully. Similarly, the options on the last parameter are for output file deletion and retention after successful transfer.

NOTE

The LPS software will not delete a raw data capture file if it is stopped manually or if an error causes processing to terminate abnormally.

E.5.2 Stopping L0R Processing

To stop L0R processing from an IRIX shell:

1. Determine the process group ID for the L0R processing you invoked by typing the following command:

    ```
    % ps -ef | grep mac_startl0R | grep -v grep
    ```

    The output will be a line that resembles the following:

    ```
    lps_op 5206 4602 0 17:51:52 pts/0 0:00 mac_startl0r 2354766 1 0
    ```

    The process group ID is the second item on the line (5206 in the example).

NOTE

If multiple L0R processes are executing on the LPS string, the command in step 1 will produce a line for each instance of L0R processing. The last item in the line is the command you used to invoke L0R processing. You can identify the process group ID of the L0R processing of interest by finding the contact sequence ID of the contact (appearing immediately after "mac_startl0r").

You can also determine the process group ID from the status and error messages output by the L0R process. The process group ID appears after the source and before the unit name. For example, in the message `Apr 2 15:36:22 1Q:lps001 syslog: 3475 [mfp_Main.c:208] CAL file could not be opened`, 3475 is the process group ID.

2. In an IRIX shell, type the following:

    ```
    % mac_stopl0r process group ID
    ```

    The L0R process will output a series of status and error messages that can be monitored to verify that the shutdown has occurred (see Section 5.4).

E.5.3 L0R Processing Parameters Management

You can view L0R process parameters and manually update both the IAS parameters and other L0R parameters through ORACLE SQL*Plus.

E.5.3.1 Viewing L0R Process Parameters

NOTE

> Each LPS string has its own set of parameters. Only the LPS string's own set of parameters may be viewed from an IRIX shell on that string.

To view the current value of any set of L0R processing parameter from ORACLE SQL*Plus:

1. Find the table and attribute names for each L0R processing parameter you want to view in Table 5–1.

2. In SQL*Plus, type the following:

```
SQL> SELECT Attributes FROM Tables
2    [ WHERE insertion_time IN
3      (SELECT MAX(insertion_time) FROM valid_ccsds_parms) ] ;
```

Attributes is a comma-separated list of the attribute names found in step 1. *Tables* is a comma-separated list of the table names found in step 1. The WHERE clause is required if, and only if, "valid_ccsds_parms" is in *Tables*.

Example: To view the values of the frame synchronization search tolerance and the value of the ETM+ processing subinterval delta time:

1. From Table 5–1, the search tolerance has the table name "valid_ccsds_parms" and the attribute name "cadu_search_tol." The subinterval delta time has the table name "valid_mfp_parms" and the attribute name "sub_intv_delta."

2. In SQL*Plus, type the following:

```
SQL> SELECT cadu_search_tol, sub_intv_delta
2 FROM valid_ccsds_parms, valid_mfp_parms
2    WHERE insertion_time IN
3      (SELECT MAX(insertion_time) FROM valid_ccsds_parms);
```

E.5.3.2 Updating IAS Parameters Manually From the IRIX Shell

IAS parameters normally should be updated when ingesting the IAS CPF.

To update the value of any IAS parameter from ORACLE SQL*Plus:

1. Find the table and attribute name for the parameter you want to update in Table 5–1.

2. Verify that the parameter's new value is within its valid range as listed in Table 5–2.

3. In SQL*Plus, type the following:

```
SQL> UPDATE Table SET Attributes = Value;
```

Table is the table name found in step 1. *Attribute* is the attribute name found in step 1. *Value* is the new value to set the parameter to.

4. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

6. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

Example: To set the value of the IAS parameter file version number to B01:

1. From Table 5–1, the IAS parameter file version number has the table name "lps_configuration" and the attribute name "IAS_PARAM_FILE_VER_NUM."

2. From Table 5–2, B01 is within the valid range for IAS parameter file version number.

3. In SQL*Plus, type the following:

```
SQL> UPDATE lps_configuration SET IAS_PARAM_FILE_VER_NUM =
"B01";
1 row updated.
```

4. The message "1 row updated." indicates the update has occurred successfully.

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

6. The "Commit complete." message indicates that the changes have been saved.

E.5.3.3 Updating and Viewing the IAS Parameters Manually from the IRIX Shell – Sensor Alignments and UTC-UT1 Corrections

The sensor alignments and UTC-UT1 corrections have slightly different procedures for updating manually. The parameters are connected to titles.

To update the values of the sensor alignment parameters from ORACLE SQL*Plus:

1. Verify that the parameter's new value is within its valid range as listed in Table 5–2.

2. View the current sensor alignment values and names in SQL*Plus by typing the following:

```
SQL> SELECT * FROM valid_sensor_align_parms;
```

This gives the full list of parameter values and titles.

3. In SQL*Plus, type the following (the single quotes are important to use):

```
SQL>UPDATE valid_sensor_align parms
2      SET sensor_align_value = Value
3      WHERE sensor_align_title = 'Title';
```

4. Verify that the update occurred correctly by noting that the following message appears:

```
1    row updated.
```

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

6. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

Example: To change the value of the Band 8 reverse odd alignment parameter to 76:

1. From Table 5–2, 76 is within the valid range for the sensor alignment parameters.

2. In SQL*Plus, type the following:

```
SQL> SELECT * FROM valid_sensor_align_parms;
```

and see that the relevant title is Reverse_Odd_B8 (note that capitalization is important). This may also be deduced from the pattern formula given in Table 5–1.

3. In SQL*Plus, type the following:

```
SQL>UPDATE valid_sensor_align parms

2       SET sensor_align_value = 76

3       WHERE sensor_align_title = 'Reverse_Odd_B8';

1 row updated.
```

4. The message "1 row updated." indicates the update has occurred sucessfully.

5. Save the change by typing the following to SQL*Plus:

```
SQL>Commit;

Commit complete.
```

6. The "Commit complete." message indicates that the changes have been saved.

To update the UTC-UT1 corrections, follow a procedure similar to that for the sensor alignment parameters:

1. To see all the values in the UTC-UT1 table, type the following in SQL*Plus:

```
SQL>SELECT * FROM UTC_UT1_Corrections;
```

This reveals the two columns as the titles of Julian dates to be CORRECTION_DATE and the correction values to be UT1_CORRECTIONS.

2. Because nominally the table contains 180 values, if you have a specific Julian date of which you want to see the current value, type the following in SQL*Plus:

```
SQL> SELECT * FROM UTC_UT1_Corrections

   2  WHERE CORRECTION_DATE = Date;
```

Remember, the DATE is the Julian date.

3. To change the value of the UTC-UT1 correction type, the following in SQL*Plus:

```
SQL> UPDATE UTC_UT1_Corrections
   2  SET UT1_Corrections = Value
   3  WHERE Correction_Date = Date;
```

4. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

6. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

E.5.3.4 Updating Other L0R Parameters Manually From the IRIX Shell

To update the value of non-IAS parameters from ORACLE SQL*Plus:

1. Find the table and attribute name for the parameter you want to update in Table 5–1.

2. Verify that the parameter's new value is within its valid range as listed in Table 5–3.

3. In SQL*Plus, type the following:

```
SQL> UPDATE Table SET Attributes = Value
   2    [ WHERE insertion_time IN
   3      (SELECT MAX(insertion_time) FROM valid_ccsds_parms) ] ;
```

   Table is the table name found in step 1. *Attribute* is the attribute name found in step 1. *Value* is the new value to set the parameter to. The WHERE clause is required if, and only if, the *Table* is "valid_ccsds_parms."

4. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

6. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

Example: To set the value of the CCSDS frame synchronization search tolerance to 3:

1. From Table 5–1, the search tolerance has the table name "valid_ccsds_parms" and the attribute name "cadu_search_tol."

2. From Table 5–3, "3" is within the valid range for CCSDS frame synchronization search tolerance.

3. In SQL*Plus, type the following:

```
SQL> UPDATE valid_ccsds_parms SET cadu_search_tol = 3
2    WHERE insertion_time IN
3       (SELECT MAX(insertion_time) FROM valid_ccsds_parms);
1 row updated.
```

4. The message "1 row updated." indicates the update has occurred successfully.

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

6. The "Commit complete." message indicates that the changes have been saved.

E.5.4 L0R Error Reporting Management

To view the current value of any set of L0R error reporting thresholds from ORACLE SQL*Plus:

1. Find the table and attribute names for each L0R error reporting threshold you want to view in Table 5–4.

2. In SQL*Plus, type the following:

```
SQL> SELECT Attributes FROM Tables ;
```

Attributes is a comma-separated list of the attribute names found in step 1. *Tables* is a comma-separated list of the table names found in step 1.

Example: To view the values for the BCH errors and CRC errors thresholds:

1. From Table 5–4, the BCH errors threshold has the table name "valid_rdp_thres" and the attribute name "bch_thres." The CRC errors threshold has the table name "valid_rdp_thres" and the attribute name "crc_thres."

2. In SQL*Plus, type the following:

```
SQL> SELECT bch_thres, crc_thres FROM valid_rdp_thres;
```

To modify the value of L0R error reporting thresholds from ORACLE SQL*Plus:

1. Find the table and attribute name for the parameter you want to update in Table 5–4.

2. Verify that the threshold's new value is within its valid range as listed in Table 5–5.

3. In SQL*Plus, type the following:

```
SQL> UPDATE Table SET Attribute = Value
```

Table is the table name found in step 1. *Attribute* is the attribute name found in step 1. *Value* is the new value to set the parameter to.

4. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

6. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

Example: To set the value of the CRC threshold to 200:

1. From Table 5–4, the CRC threshold has the table name "valid_rdp_thres" and the attribute name "crc_thres."

2. From Table 5–5, "200" is within the valid range for the CRC threshold (0....2147483648).

3. In SQL*Plus, type the following:

```
SQL> UPDATE valid_rpd_thres SET crc_thres = 200;
row updated.
```

4. The message "1 row updated." indicates the update occurred correctly.

5. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

6. The "Commit complete." message indicates that the changes have been saved.

E.6 File Transfers to EDC DAAC

This section describes how to control LPS file transfers to EDC DAAC through an IRIX shell or SQL*Plus.

- Section E.6.1 describes how to access LPS/EDC DAAC communications parameters through SQL*Plus.

- Section E.6.2 describes how to set LPS/EDC DAAC communications parameters through SQL*Plus.

- Section E.6.3 describes how to control automatic transmission of DANs to EDC DAAC through SQL*Plus.

- Section E.6.4 describes how to determine the status of the DDN server through IRIX.

- Section E.6.5 describes how to stop and then reactivate the DDN server through IRIX.

- Section E.6.6 describes how to check whether EDC DAAC has been notified.

- Section E.6.7 describes how to determine the transfer status of a set of L0R output files through SQL*Plus.

- Section E.6.8 describes how to determine through IRIX whether EDC DAAC is the process of transferring files.

- Section E.6.9 describes how to access the retention status of a file set through SQL*Plus.

- Section E.6.10 describes how to mark file sets for retention and how to delete output files through IRIX and SQL*Plus.

- Section E.6.11 describes how to delete L0R output files.

E.6.1 Viewing LPS/EDC DAAC Communication Parameters

To view the current value of the LPS/EDC DAAC communication parameters,

1. Find the table and attribute names of the parameters you wish to view in Table 6–1.

2. Type the following to ORACLE SQL*Plus:

   ```
   SQL> SELECT Attributes FROM Tables;
   ```

   Attributes is a comma-separated list of the attribute names you wish to view. *Tables* is a comma-separated list of the tables containing the attributes.

Example: To view the values of the ECS user ID, ECS password,

1. For ECS user ID, the table name is "lps_configuration" and the attribute name is "ecs_user_id." For ECS password, the table name is "lps_configuration" and the attribute name is "ecs_password."

2. Type the following to ORACLE SQL*Plus:

   ```
   SQL> SELECT ecs_user_id, ecs_password
   2>    FROM lps_configuration
   ```

E.6.2 Setting LPS/EDC DAAC Communication Parameters

To modify an LPS/EDC DAAC connection parameter from ORACLE SQL*Plus:

1. Find the attribute and table name of the parameter you wish to modify in Table 6–1.

2. Type the following to SQL*Plus:

   ```
   SQL> UPDATE Table SET Attribute = Value;
   ```

   Table and *Attribute* are the attribute names found in step 1. *Value* is the new value you want to assign to the parameter.

3. Verify that the update occurred correctly by noting that the following message appears:

   ```
   1 row updated.
   ```

4. Save the change by typing the following to SQL*Plus:

   ```
   SQL> COMMIT;
   ```

5. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

Example: To set the ECS user ID to "LPS_String1":

1. From Table 6–1, the ECS user ID table is "lps_configuration" and the attribute is "ecs_user_id."

2. Type the following to SQL*Plus:

```
SQL> UPDATE lps_configuration SET ecs_user_id = 'LPS_String1';
1 row updated.
```

3. The message "1 row updated." indicates the update has occurred successfully.

4. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

5. The "Commit complete." message indicates that the changes have been saved.

E.6.3 Automatic File Availability Notification Management

To disable automatic notification to EDC DAAC of output file availability:

1. Type the following to SQL*Plus:

```
SQL> UPDATE ldt_dan_transfer_state SET transfer_state = 0;
```

2. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

3. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

4. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

To enable automatic notification of EDC DAAC of output file availability:

1. Type the following to SQL*Plus:

```
SQL> UPDATE ldt_dan_transfer_state SET transfer_state = 1;
```

2. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

3. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

4. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

To determine the status of automatic file availability notification, type the following to SQL*Plus:

```
SQL> SELECT transfer_state FROM ldt_dan_transfer_state;
```

If the reported value of transfer_state is 0, file availability notification is disabled. If the reported value is 1, file availability notification is enabled.

E.6.4 Checking Whether the DDN Server is Active

To determine whether the DDN server is active, type the following to an IRIX shell:

```
% ps -ef | grep rcvddn | grep -v rcvddn
```

If the DDN server is active, this command will output a line that looks like

```
30 S 3098 5992 5991 0 36 24 * 3575:924 8f60e434 ttyd1 rcvddn
```

The command generates no output if the DDN server is not active.

E.6.5 Stopping the DDN Server

To stop the DDN server, type the following to an IRIX shell:

```
% killall -TERM rcvddn
```

The killall command produces no output. Follow the instructions in Section E.6.4 to verify that you have successfully stopped the DDN server.

E.6.6 Checking Whether EDC DAAC Has Been Notified

See Section 6.5.1.

E.6.7 Checking Whether EDC DAAC Has Transferred Files

See Section 6.5.2.

E.6.8 Checking Whether EDC DAAC is Transferring Files

See Section 6.6.

E.6.9 Marking a File Set for Retention

To mark a set of output files for retention:

1. Find the LPS database contact ID and the file version number for the processing instance of interest. To learn how to do so, see Sections E.1.1 and E.1.2.

2. Mark the file set for retention by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_set_info SET retention_status = 1
2>    WHERE contact_sequence_id = Contact Identifier
3>    AND file_version_number = File Version Number;
```

3. Verify that the update occurred correctly by noting that the following message appears:

```
1 row updated.
```

4. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
```

5. Verify that the changes have been saved by noting that the following message appears:

```
Commit complete.
```

Example: Mark for retention the file set produced by the first instance of processing the contact scheduled to start on day 211 of 1998 at 09:07:36.

1. Using one of the methods described in Section E.1.2, the contact ID is determined to be 86214. The file version number is 0 because this is the first instance of processing the contact.

   Mark the file set for retention by typing the following to SQL*Plus:

```
SQL> UPDATE ldt_file_set_info SET retention_status = 1
2>    WHERE contact_sequence_id = 86214
3>    AND file_version_number = 0;
1 row updated.
```

3. The message "1 row updated." indicates that the update was successful.

4. Save the change by typing the following to SQL*Plus:

```
SQL> COMMIT;
Commit complete.
```

5. The "Commit complete." message indicates that the changes have been saved.

E.6.10 Determining Whether a File Set Has Been Marked for Retention

To determine whether a file set has been marked for retention:

1. Find the LPS database contact ID and the file version number for the processing instance of interest. To learn how to do so, read Sections E.1.1 and E.1.2.

2. Type the following to SQL*Plus:

```
SQL> SELECT retention_status FROM ldt_file_set_info
2>    WHERE contact_sequence_id = Contact Identifier
3>    AND file_version_number = File Version Number;
```

3. If the value reported is 1, the file has been marked for retention. If the value reported is 0, the file has not been marked for retention.

Example: Determine whether the file set in the previous example has been marked for retention.

1. As noted in the previous example, the contact ID is 86214 and the file version number is 0.

2. Type the following to SQL*Plus:

```
SQL> SELECT retention_status FROM ldt_file_set_info
2>    WHERE contact_sequence_id = 86214
3>    AND file_version_number = 0;
PROCESSING_VERSION_INFO
-----------------------
                      0
```

3. The value reported is 0, so the file set has not been marked for retention.

E.6.11 Deleting L0R Output Files

See Section 6.9.

E.7 Report Generation

The process of generating a report is tied in with the LPS GUI in Release 2 and cannot occur through the IRIX shell.

E.8 Testing the LPS

To send test data to the LPS through an IRIX shell, invoke the LPS transmit program (rdc_Transmit) by typing the following command:

```
% rdc_Transmit [-i] [-s] filename
```

where the `filename` option is needed to specify the data file to be transmitted. The -i option is to isolate and restrict the processor and the -s option is to suspend L0R processing during transmit. These options must be invoked to happen.

Appendix F. LPS Trouble File Format

F.1 Introduction

This appendix describes the format of the LPS trouble files. The LPS design currently specifies the output of two trouble files as follows.

- The raw data processing subsystem (RDPS) produces a trouble file containing a sequence of CADUs with quality annotations. A CADU is placed in the trouble file if the CADU fails the CRC check.

- The major frame processing subsystem (MFPS) produces a trouble file containing a sequence of CADU sets such that each set is a sequence of annotated CADUs together with MFPS quality annotation representing a candidate major frame or a set of CADUs that cannot be interpreted as a major frame.

F.2 GTEDIT

The GTEDIT program is an unconfigured general-purpose binary file reader and editor that appears to be a useful tool for viewing trouble files. It can advance record-by-record through a file of variable length records if given a position and width for a binary record size field.

To use GTEDIT, read the GTSIM Users Guide. Note that the gtedit -h command gives online help. To start GTEDIT, type the following:

1. gtedit -v <filename>

2. Answer the location of record prompt with 0

3. Answer the length of record size prompt with 4

F.3 Trouble File Formats

This section contains the format for each LPS trouble file. Each trouble file includes a header record followed by zero or more data records. The structure of the header record is identical for each trouble file. Data records differ for each trouble file, but all records (both header and data) contain a size field usable by GTEDIT.

Further details of the recommended formats are as discussed in the following subsections.

F.3.1 Header

Table F–1 contains the trouble file header record format. It contains the minimum information necessary to determine the source raw data file and processing environment. Note that this format assumes trouble files are relatively short lived; in particular, it assumes additional information concerning the processing environment remains current within the LPS database on the processing string.

Table F–1. Trouble File Header Format

Name	Type	Description
Record Size	4-byte integer	Binary length of header minus 1 exclusive of this field in bytes; used by GTEDIT to traverse file
Raw Data Path/File Name	string	Name of path/file containing raw wideband data
Processing Version Number	char[2]	ASCII representation of processing version number of processing run that produced this trouble file
LPS String ID	char[20]	ASCII representation of LPS string identifier for string on which file was created

F.3.2 RDPS Data Records

Table F–2 contains the RDPS data record format. The annotated CADU structure that comprises the bulk of the data record format is defined in Table 5–1 of *LPS Interface Definitions Document* (Reference 13). Further information can be found in the global include file, lps_constants.h.

Table F–2. RDPS Data Records

Name	Type	Description
Record Size	4-byte integer	Binary length of annotated CADU entry minus 1 exclusive of this field in bytes; used by GTEDIT to traverse file
Annotated CADU	Annotated CADU structure	Binary copy of annotated CADU.

F.3.3 MFPS Data Records

Table F–3 contains the recommended MFPS data record format. The value of the specified state is given in parentheses.

Table F–3. MFPS Data Record Format

Name	Type	Description
Record Size	4-byte integer	Binary length of entry minus 1 exclusive of this field in bytes; used by GTEDIT to traverse file
General Quality	1-byte integer	Binary code indicating either bad major frame (112) or not distinguishable as a major frame (111)
Start Flag	1-byte integer	Binary code indicating record was started by either contact start (101), contact end (106), virtual channel identifier (VCID) change (110), or scan direction change (103)
End Flag	1-byte integer	Binary code indicating record was ended by either contact start (101), contact end (106), VCID change (110), or scan direction change (109)
SLsync Flag	1-byte integer	Binary code indicating whether or not line sync was detected successfully (success = 1, failure = 0)
GMT flag	1-byte integer	Binary code indicating whether or not time is correct or in error (correct = 1, error = 0)
EOL flag	1-byte integer	Binary code indicating whether or not EOL code was detected (detected = 1, not detected = 0)
Number of CADUs	2-byte integer	Binary count of number of annotated CADUs following in entry
Annotated CADUs	Annotated CADU structure	Annotated CADUs making up the tossed VCDU set

Appendix G. Support of DES by the LPS

G.1 Introduction

This appendix describes LPS support of the DES interface. Support of the DES interface was implemented with Release 3 of LPS software. The LPS now supports two operational modes: DES and EDC DACC (also known as ECS mode). When running the LPS GUI, the current mode is displayed in the lower left of the main panel as either ECS or DES. In both modes, the principal processing functions of the LPS remain the same; the only major operational change is where the LPS output products are sent and how. In ECS operational mode, LPS acts the same as the previous software release. In DES operational mode, LPS transfers the L0R, browse, and metadata files to StorageTek D3 tapes for transfer to DES. The LPS also makes browse files and Q&A data available to DES electronically.

NOTE: Until D3 tape drives are added to the LPS processing strings, LPS software should not be run in DES operational mode.

L0R data files are transferred from the LPS to DES via D3 tapes, which are prenumbered with a tape inventory number. The LPS transfers the L0R files in hierarchical data format-Earth Observing System (HDF-EOS) format and writes the files to tape as UNIX tar files, one tar file per subinterval. The LPS writes the tape inventory number to the first file on each tape. DES is responsible for maintaining the archive of D3 tapes. Tapes are transferred manually between the LPS and DES operators after writing the last contact to the tape.

G.2 Initial Setup

To run in DES operational mode, changes need to be made in the LPS database to the lps_configuration table before starting LPS. Table G–1 shows two new parameters that have been added to lps_configuration. Both the Output Mode and Tape Drive ID need to be set before proceeding further. The Output Mode needs to be set only when switching from ECS mode to DES mode (or vice versa). The Tape Drive ID only needs to be set once at initial installation.

Table G–1. New Parameters for lps_configuration

Parameter	Table Name	Attribute Name(s)
Output Mode 0 = ECS 1 = DES	lps_configuration	Outmode
Tape drive ID	lps_configuration	Tape_Device_Id

G.2.1 Set DES Output Mode

To set Output Mode to DES mode, the user must use SQL*Plus.

1. In SQL*Plus, type the following:

   ```
   SQL> UPDATE lps_configuration SET Outmode = 1;
   ```

2. Verify that the update occurred correctly by noting that the following message appears:

   ```
   1 row updated
   ```

3. Save the change by typing the following to SQL*Plus:

   ```
   SQL> COMMIT;
   ```

4. Verify that the changes have been saved by noting that the following message appears:

   ```
   Commit complete.
   ```

If there is a need to change the mode back to ECS, the above changes can be made setting outmode to zero (0).

G.2.2 Set the Tape Drive ID

Device names are automatically selected by the LPS when the hardware is hooked up. To find the information in question for a particular string:

1. Type the following on the shell command line

   ```
   hinv -c tape
   ```

   This command will give the controller and device numbers for the tape device attached to the string in question.

2. In SQL*Plus, type the following:

   ```
   SQL> UPDATE lps_configuration SET Tape_Device_Id = Value;
   ```

3. Verify that the update occurred correctly by noting that the following message appears:

   ```
   1 row updated
   ```

4. Save the change by typing the following to SQL*Plus:

   ```
   SQL> COMMIT;
   ```

5. Verify that the changes have been saved by noting that the following message appears:

   ```
   Commit complete.
   ```

6. The value for Tape_Device_Id is a string of the following format:

   ```
   /dev/rmt/tpsAdBnr
   ```

   where A is the controller number, B is the device number, and nr needs to be specified so that the tape does not automatically rewind. Note that single quotes are required.

G.3 Loading a Tape into the Tape Drive

Before loading a D3 tape in a tape drive, the Tape ID should be noted for further reference for entering in the Mount DES Tape dialog as detailed in Section G.4.2.1. A tape may be entered in the tape drive before or after entering information in the LPS GUI. The key matter is to insert the tape before LPS processing of data starts.

To load a D3 tape into the tape drive:

1. Select the proper tape drive attached to the string in question.

2. Push in the tape cartridge.

3. Slide the door down.

The display above the drives should read "loading."

4. If the display changes to "NT READY U," push the Ready button

The display above the drives should read "READY U." A U indicates that the tape cartridge is unprotected and data can be written to the tape. If the display indicated an F instead of a U, the tape cartridge would have been protected and should not be used.

NOTE: Do not unload the tape drive after it finishes writing data. This should always be done from the GUI.

G.4 LPS DES GUI Detailed Reference

This section explains each LPS GUI command and screen that has been added for the DES interface to the LPS. Operation of the DES interface to the LPS should only be done through the GUI with the exception of the commands already detailed. Two new GUI commands may be accessed through the Control Menu.

G.4.1 Main Menu Bar in DES Mode

When in DES mode, the Main Menu bar (Figure G–1) indicates this by displaying the word "DES" as indicated in the lower left.

G.4.2 Control Menu Additions for DES

The Control menu (Figure G–2) has two new commands that pertain to DES operations.

G.4.2.1 Mount DES Tape

Selecting this option brings up the MOUNT DES TAPE dialog (Figure G–3). This is the only way a tape should be mounted, otherwise the database could be corrupted. A tape may be entered in the tape drive before or after entering information in the LPS GUI. The key matter is to insert the tape before LPS processing of data starts. Enter the tape ID in the space provided. This tape ID must be directly read off of the tape that will be in the drive for the string in use. Clicking OK starts the mount process with the tape ID inserted in the database. Clicking CANCEL cancels the operation.
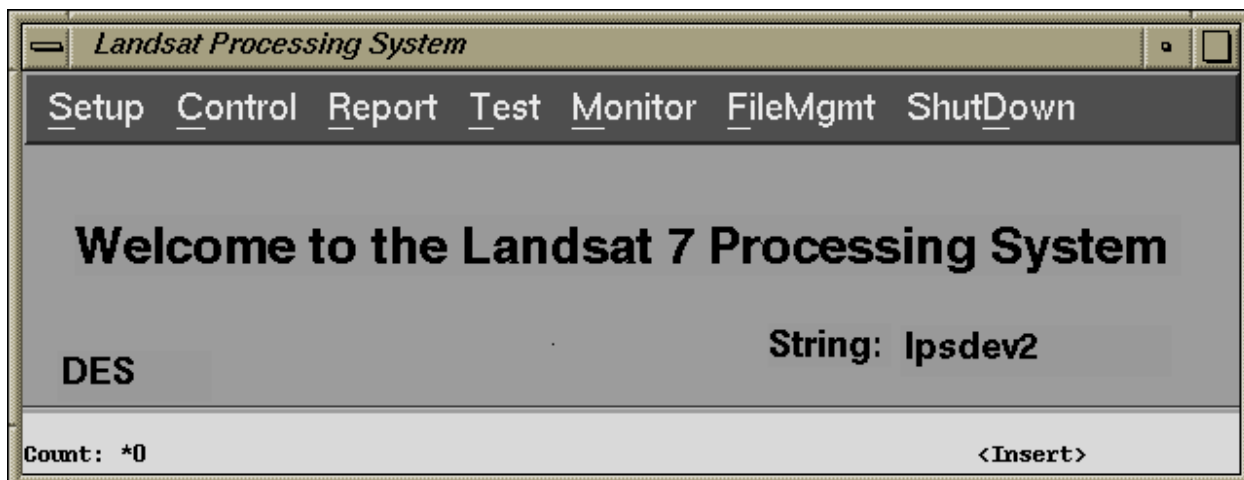
Figure G–1. LPS GUI Main Menu in DES Mode
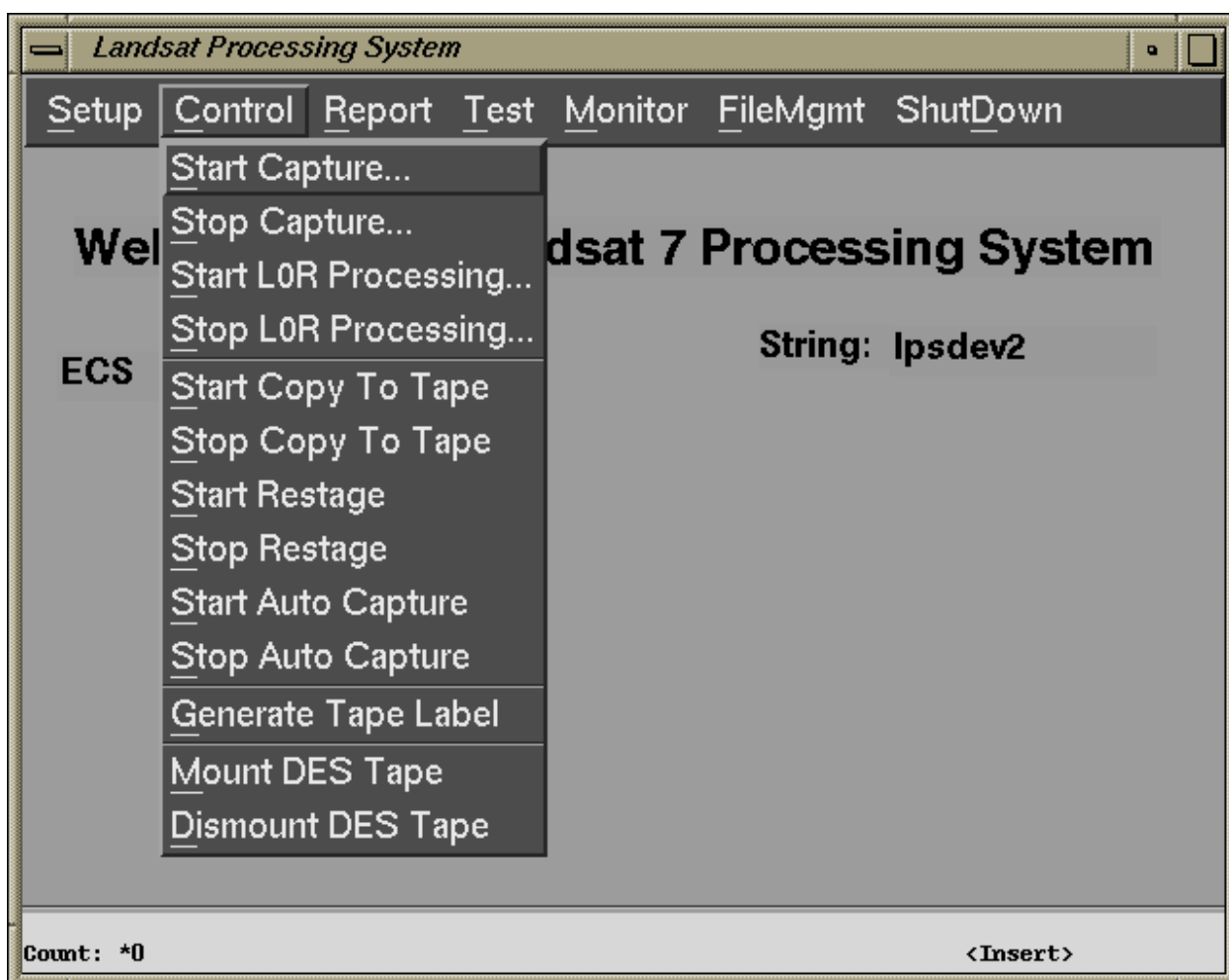
Figure G–2. LPS GUI Control Menu
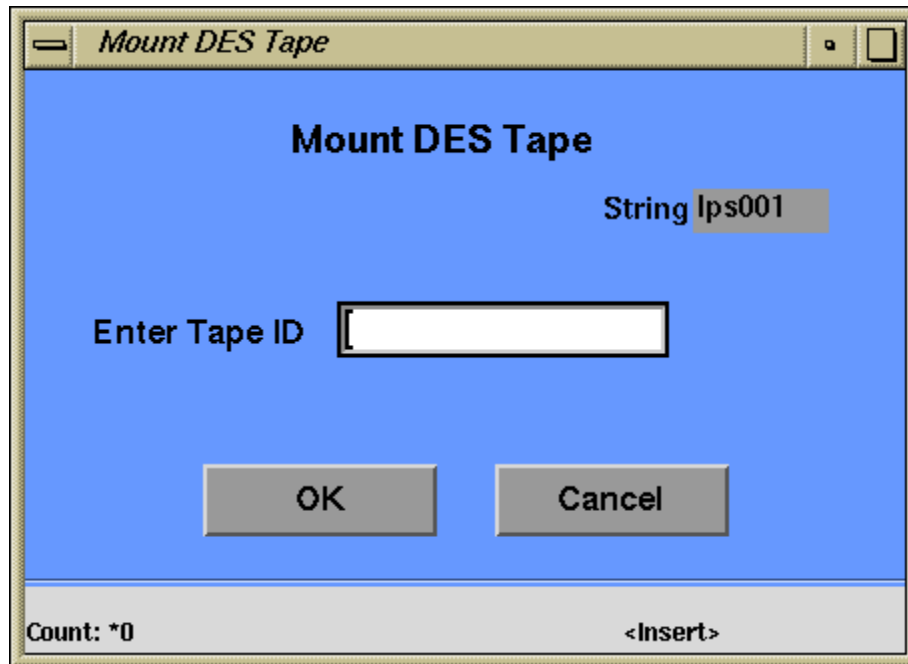
Figure G–3. LPS GUI Mount DES Tape Dialog

If a tape was already mounted in the drive attached to the string in use, a CANNOT MOUNT DES TAPE warning message (Figure G–4) will be displayed instead of the Mount DES Tape dialog. Click OK to proceed. If it is time to dismount the tape, follow the DISMOUNT DES TAPE procedure in Section G.4.2.2.

Figure G–4. LPS GUI Cannot Mount Tape Warning

If the string was not set to DES mode, a CANNOT MOUNT DES TAPE IN ECS MODE warning message (Figure G–5) will be displayed instead of the MOUNT DES TAPE dialog. Click OK to proceed. To set DES Output Mode, follow the procedure in Section G.2.1.

Figure G–5. LPS GUI Cannot Mount Tape in ECS Mode Warning

G.4.2.2 Dismount DES Tape

Selecting this option brings up the DISMOUNT DES TAPE dialog (Figure G–6). This is the only way a tape should be dismounted, otherwise the database could be corrupted. The Tape ID for the tape in the drive attached to the present string will be displayed. Click OK to proceed. Click CANCEL to stop the operation.
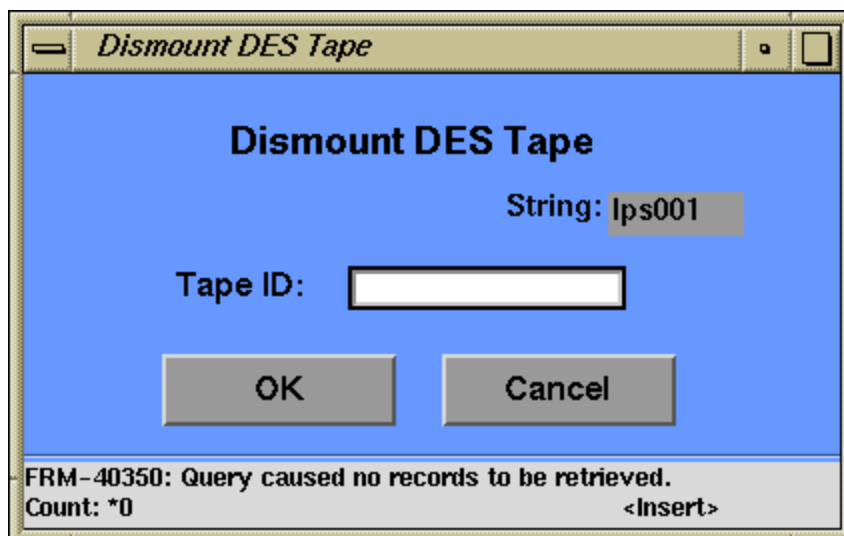
Figure G–6. LPS GUI Dismount DES Tape Dialog

If no tape is in the drive attached to the present string or the tape is busy, a CANNOT DISMOUNT DES TAPE warning message (Figure G–7) will be displayed instead of the DISMOUNT DES TAPE dialog.

Figure G–7. LPS GUI Cannot Dismount Tape Warning

If the string was not set to DES mode, a CANNOT DISMOUNT DES TAPE IN ECS MODE warning message (Figure G–8) will be displayed instead of the DISMOUNT DES TAPE dialog. Click OK to proceed.
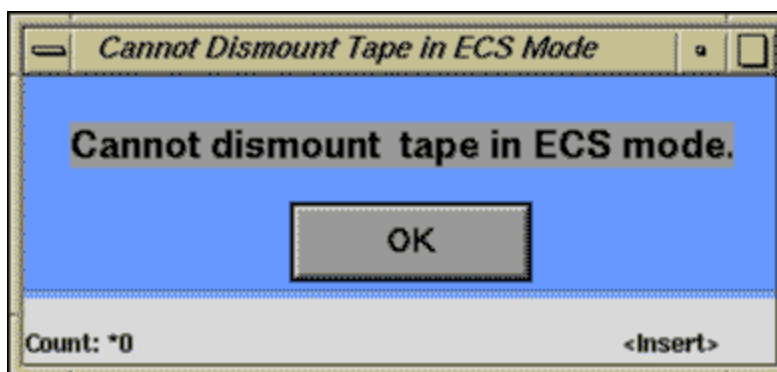
Figure G–8. LPS GUI Cannot Dismount Tape in ECS Mode Warning

G.5 Other DES Database Parameters

Two new tables have been inserted into the LPS database (Table G–2). They may be queried as needed from SQL*Plus.

Table G–2. LPS Database Tables

Parameter	Table Name	Attribute Name(s)
Contact sequence ID	DES_Info	Contact_Sequence_Id
Version number	DES_Info	File_Version_Number
L0R processing complete flag	DES_Info	L0R_Completion_Status
Subinterval ID	DES_Info	Sub_Intv_Sequence_Id
Tape inventory number	DES_Info	Tape_Id
Position on tape – number of end of files to skip to get to this subinterval	DES_Info	Sub_Intv_Position
Received by DES flag	DES_Info	DES_Receive_Status
Tape number currently mounted	SD3_Tape_Info	Tape_Id
Tape drive's in-use status	SD3_Tape_Info	Tape_Status
Number of subintervals written to tape	SD3_Tape_Info	Sub_Intv_Count

Abbreviations and Acronyms

ACCA	Automated Cloud Cover Assessment
ASCII	American Standard Code for Information Interchange
BCH	Bose-Chaudhuri-Hocquenghem (error detection and correction)
CADU	channel access data unit
CCA	Cloud Cover Assessment
CCR	configuration change request
CCSDS	Consultative Committee for Space Data Systems
CNMOS	Consolidated Network Management and Operations Support
CPF	calibration parameter file
CRB	Central Review Board
CRC	cyclic redundancy code
DAAC	Distributed Active Archive Center
DAA	data availability acknowledgment
DAN	data availability notice
DBA	database administrator
DBMS	database management system
DCN	document change notice
DDA	data delivery acknowledgment
DDN	data delivery notice
DES	DAAC Emergency System
DLT™	Digital Linear Tape
ECS	EOS Core System
EDC	EROS Data Center
EOL	end of line
EOS	Earth Observing System
ETM+	Enhanced Thematic Mapper Plus
EROS	Earth Resources Observation System

FDDI	fiber-distributed data interface
FTP	File Transfer Protocol
GB	gigabyte
GTSIM	Generic Telemetry Simulator
GUI	graphical user interface
HDF	hierarchical data format
IAS	Image Assessment System
ID	identifier
IDPS	image data processing subsystem
IFOV	instantaneous field-of-view
IP	Internet Protocol
L0R	Level Zero R
LGS	Landsat Ground Station
LPS	Landsat 7 Processing System
MACS	management and control subsystem
Mbps	megabits per second
MFPS	major frame processing subsystem
MOC	Mission Operations Center
MSCD	mirror scan correction data
MWD	Moving Window Display
PCD	payload correction data
Q&A	quality and accounting
RAID	redundant arrays of independent disks
RAM	random access memory
RDPS	raw data processing subsystem
SCID	spacecraft identifier
SGI	Silicon Graphics, Inc.
SQL	Structured Query Language
TBA	to be added

TBD	to be defined
TBR	to be resolved
UTC	universal time coordinated
VCDU	virtual channel data unit
VCID	virtual channel identifier
VME	Versa-Module European
WRS	Worldwide Reference System

Glossary

Band file | A **Level 0R output file** containing the image data from a single band in a single subinterval.

Browse image file | A **Level 0R output file** containing a reduced resolution image in three spectral bands of a scene.

Calibration file | A **Level 0R output file containing** the calibration data collected at the end of each scan for a single subinterval.

Contact (period) | The time period between the start and end of wideband data transmissions from the Landsat 7 satellite to a ground station.

Daemon (process) | A process without a controlling terminal that executes in the background. LPS daemon processes perform automatic **data capture**, copying to tape, **Level 0R processing**, and **DDN** processing. The IRIX **syslogd** daemon formats and routes LPS status and error messages to the **LPS Journal file**.

Data availability acknowledgment (DAA)

An LPS/EDC DAAC communication message sent by EDC DAAC to acknowledge receipt of a **data availability notice**.

Data availability notice (DAN)

An LPS/EDC DAAC communication message sent by the LPS to notify EDC DAAC that a **file set** is available for transfer.

Data delivery acknowledgment (DDA)

An LPS/EDC DAAC communication message sent by the LPS to acknowledge receipt of a **data delivery notice**.

Data delivery notice (DDN)

An LPS/EDC DAAC communication message sent by EDC DAAC to notify the LPS of the disposition of its attempts to transfer a **file set**.

DDN server | An LPS **daemon** that accepts and processes **DDNs** from EDC DAAC.

Enhanced Thematic Mapper Plus (ETM+)

The imaging instrument onboard Landsat 7.

ETM+ format | One of two 150-Mbps telemetry streams output by the ETM+. Format 1 contains spectral Bands 1 through 6. Format 2 contains spectral Bands 6, 7, and 8. Both formats include calibration data , mirror scan correction data, and payload correction data.

File group	A collection of **Level 0R output files** for a single subinterval comprising one component of **file set**.
File set	A collection of all **Level 0R output files** for a single **Level 0R processing** run. The file set includes **file groups**, one for each subinterval extracted by the processing.
I channel	A 75-Mbps portion of an **X-band** channel containing either **format 1** or **format 2** telemetry data.
IRIX	A version of the UNIX operating system running on **LPS strings** and Indy **workstations**.
Level 0R output files	The set of files produced by L0R processing, consisting of **band files**, **browse files**, **calibration files**, **MSCD files**, and **metadata files.**
Level 0R processing	The operation of extracting Landsat 7 images, as well as correction and calibration information from **raw wideband data**, to produce **Level 0R output files**.
Level 0R Q&A data	The data Q&A information collected by the LPS from processing of the ETM+ major frames during **Level 0R processing**.
LPS database	A persistent storage repository containing configuration elements that can be set, **Level 0R processing** parameters and error reporting thresholds as well as Q&A information for **data capture**, **Level 0R processing**, and **Level 0R output file** transfer.
LPS GUI	The LPS graphical user interface that provides access to LPS functions on a single **LPS string**. The LPS GUI consists of a main menu from which LPS commands can be selected, additional dialogs for providing parameters to the command, and confirmation dialogs to prevent accidental command execution.
LPS Journal file	An ASCII file containing all of the status and error messages generated by the LPS software.
LPS string	One of five computers that hosts LPS processing. Four LPS strings are operational and perform **data capture** (each captures a single 75-Mbps channel) and **Level 0R processing** functions. A fifth string is used for training and development and as an emergency backup.
Metadata file	A **Level 0R output file** containing information on the L0R data provided in the subinterval and the names of the L0R **band file**, **calibration file**, **PCD file**, **MSCD file,** and **browse image files** associated with the subinterval. Metadata also contains Q&A information on the data contained in the subinterval.
Mirror scan correction data (MSCD) file	
	A **Level 0R output file** containing the first half scan error, second half scan error, and scan direction for each scan.

ORACLE   A commercial DBMS used to manage the **LPS database** on each **LPS string**.

Payload correction data (PCD) file

                           A **Level 0R output file** containing PCD major frames received during a subinterval.

Playback (telemetry)   Telemetry data that is stored onboard the satellite when it is generated and replayed to a ground station at a later time.

Q channel   A 75-Mbps portion of an **X-band** channel containing either **format 1** or **format 2** telemetry data.

Raw wideband data   Telemetry data as received from Landsat 7 through the Landsat Ground Station.

Real-time (telemetry)   Telemetry data that is transmitted to a ground station as it is generated by the ETM+.

Retention status   Whether or not a **Level 0R output file** is marked in the **LPS database** as to be deleted automatically after successful transfer to EDC DAAC.

Return-link Q&A data

                           The data Q&A information collected by LPS L0R processing from CCSDS Grade 3 and BCH error detection and correction processing during **L0R processing**.

Shell   One of several IRIX programs that provide a command line interface for invoking programs, built-in commands for performing a set of operations such as echoing input, and an environment in which variables controlling program operation can be assigned persistent values. Shells also understand rudimentary flow control and will read script files containing commands. This provides a programming capability. Popular shells include the C shell [csh(1)] and Bourne shell [sh(1)].

SQL*Plus   The **ORACLE** SQL interpreter.

Subinterval   A segment of continuous **raw wideband data** received during a Landsat 7 **contact period**. Subintervals are caused by a shift from real-time to playback data or by a shift from one segment of playback to another collected at different time and location. Subintervals may also occur because of breaks in the wideband data stream due to communication dropouts. The largest possible subinterval can be as long as a full contact. The smallest possible subinterval can be as small as one full **WRS scene** with a duration of approximately 24 seconds.

telnet   An IRIX program that allows login and interaction with a remote host.

Workstation   An LPS hardware component used to host the **Moving Window Display** and, optionally, providing an interface to the **LPS strings**. Two

workstations are normally used to host the **Moving Window Display** with each workstation hosting the display for two **LPS strings**. A third workstation is used for controlling the IRIS console.

WRS Scene A frame of imagery defined by the Worldwide Reference System. Each scene is defined as a swath of approximately 179 kilometers (163 kilometers plus 10 percent in-track overlap) centered on a position along the Landsat 7 satellite's track. The WRS assigns sequential path numbers to the satellite's 233 nominal orbit tracks. The WRS also defines 119 numbered latitudinal center lines, called rows, that intersect the paths (row 60 is at the equator). WRS scene centers are defined as the points of intersection of the path and row lines. The scenes are designated by the path and row number at the scene center.

X band A 150-Mbps downlink channel from Landsat 7. The satellite has three X-band channels. During any contact, at most two of them are active. Each X-band downlink consists of two 75-Mbps **I** and **Q channels.**

X terminal An LPS hardware component providing an interface to the **LPS strings** and **workstations**. Two X terminals normally host the **LPS GUIs** for the four operational strings, with each X terminal hosting two strings.

xwsh An IRIX program that provides an X window interface to a **shell** or to a specified application program.

References

1. *LPS Programmers Reference Manual*, 514-30MM/0196, July 23, 1997

2. *IRIX 5.3 Reference Manual*, publishing information TBA

3. *LPS Operations and Maintenance Manual*, 514-3OMM/0196, November 27, 1996

4. *LPS Installation Procedure*, 514-2IP/0195, October 14, 1996

5. *IRIX Essentials*, 007-1342-100, March 1995

6. *Oracle 7 Server Messages and Codes Manual*, Release 7.1, Part No. A12379-2, May 1994

7. *Oracle Developer/2000 Forms 4.5 Messages and Codes Manual*, Part No. A32508-1, 1994

8. *Oracle Developer/2000 Reports 2.5 Messages and Codes Manual*, Part No. A32490-1, April 1995

9. *Landsat 7 Processing System (LPS) Software Configuration Guide*, July 1997

10. *Oracle 7 for MIPS ABI Systems Installation and Configuration Guide*, Release 7.2, Part No. A36047-1, August 1995

11. *Oracle 7 Server SQL Reference*, Release 7.2, Part No. A20325-2, April 1995

12. *EDC Landsat 7 DHF Standard Operating Procedures*, publishing information TBA

13. *LPS Interface Definition Document*, 513-3IDD/0195, October 1996

Index